UNIVERSIDADE FEDERAL DO PARÁ

INSTITUTO DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

PROJETO E IMPLEMENTAÇÃO DE UM SISTEMA DE AUTORIA PARA ANIMAÇÕES, SIMULAÇÕES E TREINAMENTOS EM REALIDADE VIRTUAL

PEBERTLI NILS ALHO BARATA

DM: 17 / 2010

UFPA / ITEC / PPGEE

CAMPUS UNIVERSITÁRIO DO GUAMÁ

BELÉM – PARÁ – BRASIL

SETEMBRO-2010

UNIVERSIDADE FEDERAL DO PARÁ

INSTITUTO DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

PROJETO E IMPLEMENTAÇÃO DE UM SISTEMA DE AUTORIA PARA ANIMAÇÕES, SIMULAÇÕES E TREINAMENTOS EM REALIDADE VIRTUAL

PEBERTLI NILS ALHO BARATA

DISSERTAÇÃO SUBMETIDA À BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA DO INSTITUTO DE TECNOLOGIA DA UNIVERSIDADE FEDERAL DO PARÁ, COMO REQUISITO PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE COMPUTAÇÃO APLICADA.

UFPA / ITEC / PPGEE

CAMPUS UNIVERSITÁRIO DO GUAMÁ

BELÉM – PARÁ – BRASIL

SETEMBRO-2010

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

PROJETO E IMPLEMENTAÇÃO DE UM SISTEMA DE AUTORIA PARA ANIMAÇÕES, SIMULAÇÕES E TREINAMENTOS EM REALIDADE VIRTUAL

AUTOR: Pebertli Nils Alho Barata

DISSERTAÇÃO DE MESTRADO SUBMETIDA À AVALIAÇÃO DA BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA DA UNIVERSIDADE FEDERAL DO PARÁ E JULGADA ADEQUADA PARA OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE COMPUTAÇÃO APLICADA

APROVADA EM 03/09/2010

BANCA EXAMINADORA:

Prof. Dr. Manoel Ribeiro Filho
(ORIENTADOR – UFPA)

Prof. Dr. Marcus Vinícius Alves Nunes
(MEMBRO – UFPA)

Prof. Dr. Marcos Paulo Alves de Sousa
(MEMBRO – Museu Paraense Emílio Goeldi)

VISTO:

Prof. Dr. Marcus Vinícius Alves Nunes
(COORDENADOR DO PPGEE/ITEC/UFPA)

RESUMO

Esta dissertação apresenta o projeto, desenvolvimento e implementação de um software para criação de animações, simulações e treinamentos de processos e instruções técnicas industriais, destacando também a arquitetura utilizada para sua implementação. O software apresentado permite a geração de animações, textos, sons e interações, fornecendo ferramentas de criação, visualização e supervisão da simulação com o uso de Realidade Virtual em três dimensões. A dissertação trata ainda do uso prático do software através de um estudo de caso com simulações 3D de processos industriais reais de uma usina de geração de energia elétrica. Este estudo também contribui para as áreas de treinamento em realidade virtual e sistemas de autoria 3D ao apresentar conceitos inovadores na área de supervisão automática, mostrando também uma possível estrutura de componentes de software para uso nas áreas citadas. Pode-se destacar ainda a revisão de tecnologias semelhantes à proposta neste trabalho, visando à identificação de influências, melhorias e inovações nas áreas de autoria 3D e treinamento.

PALAVRAS-CHAVES: Autoria, Realidade Virtual, Simulação, Treinamento

ABSTRACT

This dissertation presents the design, development and implementation of a software for the creation of animation, simulations and training processes and industrial technical instructions, also presents the architecture used for its implementation. The software presented allows the generation of animation, texts, sounds and interactions, providing tools for creating, display and supervision of simulation with use of Virtual Reality 3D. The dissertation also deals with the practical use of the software through a case study with simulations 3D in real industrial processes of a plant for generation of electric energy. This study also contributes to the areas of training and virtual reality and systems of authorship 3D to submit innovative concepts in the area of supervision automatic, also showing a possible structure of software components for use in the areas mentioned. May also be highlighted the review of similar technologies to the purpose of this study, aiming to the identification of influences, improvements and innovations in the areas of authorship 3D and training.

KEYWORDS: Authoring, Virtual Reality, Simulation, Training

SUMÁRIO

LISTA DE ILUSTRAÇÕES

1 INTRODUÇÃO

A simulação de um processo é uma ferramenta capaz de gerar um cenário virtual equivalente ao cenário real, a fim de avaliar, melhorar ou mesmo visualizar o processo em questão. Essa ferramenta é utilizada quando o cenário real do processo não pode ser interrompido ou visualizado facilmente. Sendo assim, dentre algumas situações que uma simulação seria recomendada, estão: projetar, analisar e realizar treinamentos em operações industriais; análise de estratégias de logística; treinamento em operações de máquinas e veículos. (LAW e KELTON, 1999)

Dentre os modos de simulação mais conhecidos estão a simulação gráfica e a simulação matemática. Em específico, a simulação gráfica em realidade virtual é uma das mais usadas no mundo devido à capacidade natural que a realidade virtual tem de prover imersão no cenário simulado. Para isso, RV (Realidade Virtual) se utiliza de dispositivos computacionais que exploram diversos sentidos humanos, como visão, tato e audição. Este tipo de simulação pode desempenhar funções como suporte ao cliente, manutenção, procedimentos operacionais ou treinamento. Porém, a construção de uma simulação em RV pode não ser tão simples ou rápida para justificar o seu uso: A construção requer o conhecimento de modelagem 3D, programação de computadores, matemática 3D e conhecimento do processo que será simulado. Desta forma, em alguns casos, a construção da simulação pode representar uma dificuldade tão grande quanto o uso do cenário real. A situação ideal é que o foco de trabalho esteja na simulação em si, e não na construção da simulação, ou seja, o que se quer é uma ferramenta que possibilite criar uma simulação da maneira mais rápida, natural e transparente possível para que todos os esforços se concentrem apenas na análise da simulação.

Felizmente, existem ferramentas que possibilitam a criação de cenários em RV de maneira simples e rápida e muitas vezes sem retirar o foco no objetivo final que é a simulação. Essas ferramentas são conhecidas como sistemas de autoria. Esta dissertação apresenta a estrutura e implementação de uma solução de software para criação de simulações em RV de processos operacionais industriais, um sistema de autoria. Como pretendido, o software permite a criação de simulações sem conhecimento de programação de computadores bastando poucas horas de treinamento no software em questão.

O software implementado pela equipe de desenvolvimento do Laboratório de Realidade Virtual da UFPA (LARV) e apresentado nesta dissertação, recebe o nome de Sistema de Autoria de Instruções Técnicas Virtuais ou Sistema de Autoria ITV. E é um software de autoria para animações, simulações e treinamentos em Realidade Virtual em 3D. Porém, mais que isso, o software também permite a visualização e execução de todas as animações e treinamentos feitos nele próprio.

O projeto de software do Sistema de Autoria ITV teve início em 2008 com parceria da Eletronorte S.A. (ELETRONORTE, 2010), empresa gerenciadora da Usina Hidrelétrica de Tucuruí no estado do Pará. Uma Usina Hidrelétrica ou UHE é um conjunto de equipamentos que tem por função transformar a energia potencial da água em energia elétrica. Esta transformação ou geração pode ser resumida da seguinte forma, segundo (KIAMEH, 2002):

A água captada é conduzida através de um sistema de tubulação capaz de agregar pressão a ela. Em geral, a água encontra-se represada através de uma barragem. A pressão da água na tubulação é capaz de girar um rotor que por sua vez gira um gerador. O giro do gerador produz um campo eletromagnético que pode ser coletado e distribuído como a energia elétrica que usamos no dia-a-dia.

Em um primeiro momento, o Sistema de Autoria ITV tem o objetivo de criar e simular os diferentes processos ocorridos na UHE de Tucuruí, a exemplo de uma limpeza na sala de máquinas, manutenção do rotor ou vistoria na barragem. Cada um destes processos é descrito de maneira formal em um documento chamado de Instrução Técnica ou IT. Na UHE de Tucuruí algumas das IT's mais importantes são as IT's de Operação e IT's de Manutenção, que descrevem, respectivamente, um processo de operação de algum equipamento e um processo de reparação, limpeza ou vistoria de algum equipamento. O Sistema de Autoria ITV pode, eficientemente, descrever totalmente uma IT real em uma IT Virtual (ITV), simulando cada passo a ser seguido durante o processo regido pela IT.

O projeto proposto e implementado apresentado nesta dissertação, validou em diferentes momentos a sua utilização. Sendo assim, este trabalho também apresenta um estudo de caso onde a Instrução Técnica de Operação Partida de uma Unidade Geradora utilizada na UHE de Tucuruí foi totalmente descrita na forma de ITV com o uso do Sistema de Autoria ITV.

Além deste capítulo de introdução que descreveu resumidamente o objetivo do trabalho e a motivação de sua realização, existem ainda cinco capítulos.

O capítulo 2 apresenta trabalhos que serviram de base para a criação do software Sistema de Autoria ITV, além de mostrar sistemas de autoria semelhantes ao sistema mostrado nesta dissertação. Além de apresentar o sistema de software UGV que originou o início do projeto atual mostrado neste trabalho.

O capítulo 3 explica o desenvolvimento do Sistema de Autoria ITV enumerando alguns requisitos iniciais que compõe o software. Apresenta cada uma das tecnologias utilizadas no desenvolvimento da aplicação. Este capítulo também explica a organização da arquitetura de funcionamento do sistema.

O capítulo 4 discorre sobre o Sistema de Autoria ITV implementado, mostrando suas funções e o funcionamento principal durante a confecção de um projeto ITV. Também explica os conceitos de uma animação e treinamento ITV e o modo como ela é montada no sistema.

No capítulo 5 tem-se o estudo de caso de uso do Sistema de Autoria ITV de uma instrução técnica da UHE de Tucuruí.

O capítulo 6 apresenta algumas funções que poderiam ser agregadas ao Sistema de Autoria ITV em um projeto futuro.

2 TRABALHOS CORRELATOS

A construção de aplicações gráficas em 3D nunca foi uma tarefa simples, especialmente quando esta construção se dá através de linguagens de programação usuais como a linguagem C++ ou a programação de *scripts shaders* para placas de vídeo(RANDIMA, 2004). Além disso, muitas vezes aplicações com arquiteturas semelhantes são desenvolvidas sem a prática do reuso, ou seja, desenvolve-se a aplicação sem aproveitar componentes de software anteriormente utilizados, o que demanda um esforço desnecessário na criação da aplicação.

A indústria de jogos eletrônicos é um grande exemplo desta situação: de acordo com (DAMASCENO, 2009), o desenvolvimento de um jogo comum, necessita de enredo, arte, desenho, sonorização, modelagem 3D, projeto de jogabilidade, programação, testes e distribuição. Porém algumas destas etapas poderiam ter seus esforços suavizados a partir do reaproveitamento de componentes, o que permitiria concentrar maiores esforços em questões fundamentais para o sucesso de jogos, a exemplo de enredo e modelagem 3D.

Atualmente, existem softwares que permitem esta "suavização" de esforço em etapas ou mesmo a total extinção delas na criação da aplicação 3D. Estes softwares são conhecidos como *middleware* e são bastante populares na forma de *game engine's*. O termo *game engine* é popular desde a década de 90 com o uso de ferramentas para customização de jogos em 1ª pessoa. Atualmente o termo é usado também para designar as ferramentas de construção de simulações ou visualizações em 3D, sendo que estas ferramentas são exatamente *game engine's*, porém não são usadas para construir jogos.

Este capítulo trata da importância destas ferramentas no desenvolvimento de aplicações 3D. Trata também dos projetos anteriores que deram origem ao trabalho apresentado nesta dissertação.

2.1 Game engine's

Uma *game engine* é um software projetado para o desenvolvimento de jogos eletrônicos. A sua principal característica é a abstração de software e hardware, onde o usuário desenvolvedor do jogo irá trabalhar com uma interface amigável enquanto a *engine* se encarrega das funções do núcleo do jogo, por exemplo, renderização gráfica, gerenciamento

de memória, detecção de colisão, reprodução de som, gerenciamento de animações, grafo de cena, etc (EBERLY, 2000). Devido a esta abstração, uma *game engine* pode ser classificada como um software *middleware*, pois é a camada intermediária separando o núcleo do jogo do desenvolvedor.

O uso de uma *engine* no desenvolvimento de jogos permite a redução da complexidade, custo e tempo de desenvolvimento à medida que o desenvolvedor não precisa programar na linguagem de programação da plataforma hospedeira ou usar matemática 3D. Algumas *game engine's* nem mesmo utilizam scripts, possibilitando maior simplicidade na criação dos jogos. Por outro lado, existem *engines* que são conjuntos de bibliotecas de programação especializadas e não podem ser chamadas de *game engine*, já que não desempenham todas as funções necessárias para a criação de jogos, por exemplo, Ogre é uma *engine* gráfica 3D (OGRE, 2010) escrita para a linguagem de programação C++, então, para usar Ogre é necessário programar em C++ auxiliado pela biblioteca Ogre. Além disso, algumas *engines* desempenham funções específicas no desenvolvimento. É o caso, por exemplo, da FMOD (FMOD, 2010), uma biblioteca escrita em C++ para reprodução e gravação de som em aplicações de software multiplataforma.

Porém é ideal que uma *game engine* desempenhe todas as funções possíveis no processo de desenvolvimento, sendo que, deste modo, o criador do jogo poderá se concentrar na parte criativa do game sabendo que tudo o que ele imaginar poderá ser realizado sem maiores problemas pela *game engine*. De acordo com (GREGORY, 2009) as principais funções de uma *game engine* típica são:

- Abstração e gerenciamento de recursos de software e hardware
- Editor de cenário com grafo de cena
- Renderização gráfica 3D e 2D
- Detecção de colisão e física
- Gerenciamento de luz e sombras
- Suporte a diferentes tipos de modelos 3D, texturas e sons
- Sistema de partículas
- Inteligência artificial
- Gerenciamento de rede para multijogador

2.1.1 Estado da arte

O grande benefício das *game engine's* está na eficiência da produção, portanto não é surpresa que cada vez mais elas estejam se desprendendo de linguagem de programação ou scripts. Pode se ver hoje em dia, por exemplo, o Kodu Game Lab (KODU, 2009) que pode ser descrito como um jogo de fazer jogos; com uma linguagem de programação totalmente visual baseada em ícones. É bem verdade que uma *game engine* para desenvolvimento de jogos de qualquer gênero (tiro em primeira pessoa, corrida, luta, esporte, etc.) ainda não se livrou da programação tradicional, porém também está caminhando para isto, como é o caso da plataforma XNA (NITSCHKE, 2007), para jogos em celular, PC, e Xbox 360, que a cada versão utiliza cada vez menos programação em linguagem C# pura dando suporte total e automático a multijogador e à distribuição e venda da aplicação.

Outra grande característica das *game engine's* atuais está na melhoria do gerenciamento dos recursos de hardware e software, permitindo, por exemplo, o uso de mais de uma placa de vídeo, diminuindo a carga de processamento na CPU e aumentando na GPU, "limpando" recursos inutilizados, usando recursos novos como o *tesselation* que é uma técnica de alteração dinâmica do nível de detalhe dos modelos 3D (TATARINOV, 2008). Estas características em especial são importantes para a longevidade de uma *game engine* a qual evolui mais lentamente que a tecnologia envolvida em sua execução, ou seja, uma *game engine* criada em 2008 deve ter suporte para uma tecnologia criada em 2010.

Um fator que se torna essencial no mercado de jogos atual é a distribuição multiplataforma. O ideal é que uma *game engine* produza um jogo, da mesma forma e sem qualquer alteração no projeto, em várias plataformas. Atualmente se vê o lançamento de jogos para três ou quatro plataformas ao mesmo tempo sem muita diferença entre si. Esta característica agrada tanto desenvolvedores, que não precisaram ter um retrabalho na criação, quanto usuários dos jogos que vão poder usufruir do jogo naquela plataforma que eles dispõem ou preferem.

A generalização da *game engine* é importante para o sucesso comercial, afinal nem só de jogos em 1ª pessoa é que vive a indústria de jogos, ou seja, a engine deve permitir a produção dos mais variados gêneros de jogos não só de maneira aceitável, mas de maneira otimizada como se a *game engine* fosse especializada naquele gênero. É o caso, por exemplo, da Torque 3D (TORQUE, 2009) que possibilita o desenvolvimento fácil e otimizado de diferentes gêneros de jogos.

Pode-se citar também algumas tecnologias em ascendência que já estão sendo usadas por algumas *game engine's* como a Unreal Engine(UNREAL, 2010): suporte a *motion*

capture (tecnologia de captura de movimentos de modelos reais diretamente para o modelo 3D) e *facial motion capture* (captura dos mínimos movimentos do rosto humano para o modelo 3D), suporte a hardware escalável, inteligência artificial com aprendizado, criação dos modelos 3D na própria engine, módulo de testes automáticos, suporte aos novos periféricos sensores de movimentos e criação de jogos estilo *sandbox* o qual permite ao jogador a liberdade de explorar o cenário inteiro do jogo de forma não linear podendo fazer sua própria seqüência de objetivos para finalização do game.

2.1.2 *Exemplos de game engine's*

CryENGINE® 3

Esta *game engine*, ainda em fase final de desenvolvimento pela empresa CryTek, dará suporte as plataformas PC (DirectX 9, 10 e 11), Xbox 360 e Playstation 3 e tem como principal característica o elevado grau de definição gráfica, usando recursos como texturas de alta resolução, luz volumétrica, alta fidelidade na simulação de flúidos e HDR *lighting* (*High Dynamic Range*) (HOULMANN e METZ, 2009). O principal jogo desenvolvido com está *engine* é o já anunciado Crysis 2. A figura 2.1 mostra algumas imagens do uso da *engine*.
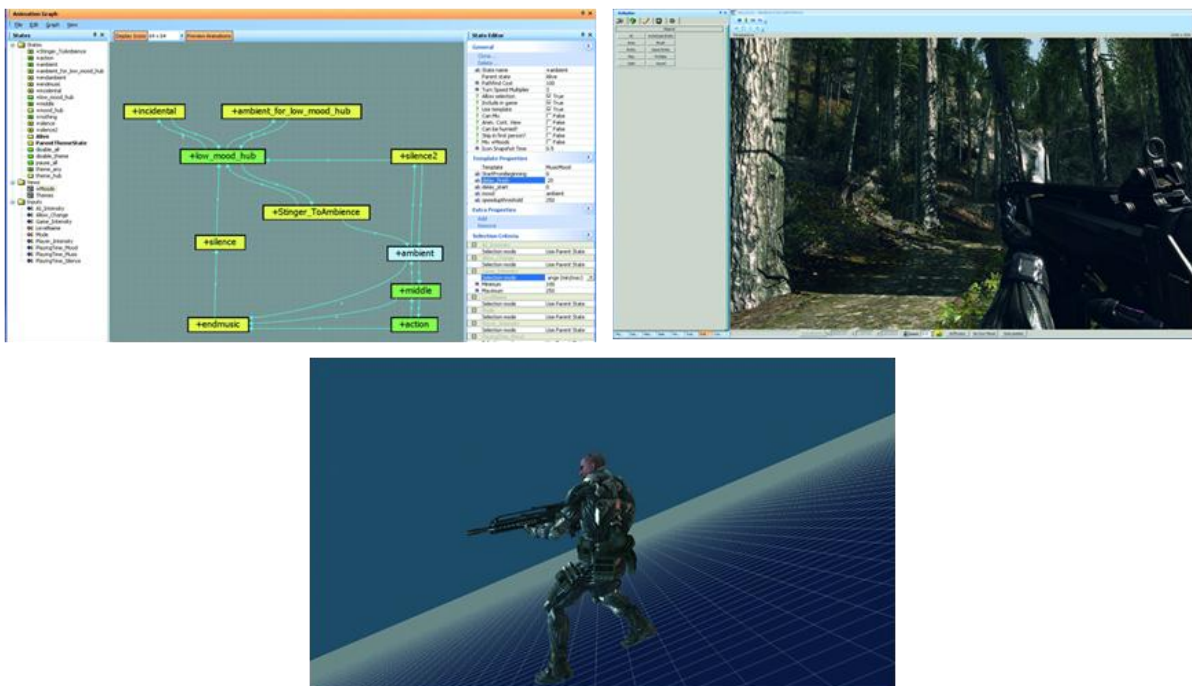
Figura 2.1 - Utilização da *CryENGINE® 3*

A *CryEngine*® 3 possui uma plataforma de desenvolvimento unificada, ou seja, qualquer alteração no desenvolvimento é refletida para todas as plataformas de distribuição (Xbox 360 e Playstation 3, por exemplo) suportadas ao mesmo tempo, fazendo automaticamente a adequação de parâmetros de performance e qualidade. Outra inovação apresentada por esta *game engine* está na Inteligência Artificial, onde os personagens possuem sentidos de visão, tato e audição, podendo perceber, por exemplo, a aproximação de inimigos somente por ouvir os passos deles.

RAGE – Rockstar Advanced Game Engine

A Rockstar Advanced Game Engine (RAGE) é uma *game engine* desenvolvida por um time de programadores da empresa Rockstar. Ela tem suporte para as plataformas PC, Xbox 360, PlayStation 3 e Nintendo Wii e é especializada em jogos do tipo *sandbox*, apesar de não ter sido criada exclusivamente para isso. É importante constatar que a RAGE possui módulos de terceiros como a *engine* física Bullet. Alguns jogos de grande sucesso foram desenvolvidos na RAGE como *Grand Theft Auto IV* além do lançamento *Red Dead Redemption.* A figura 2.2 apresenta uma imagem de um jogo feito na RAGE.

Figura 2.2 - Imagem do jogo *Read Dead Redemption*, em desenvolvimento na *engine* RAGE

A *RAGE* gerencia o carregamento de recursos de uma forma única até a sua criação: a *engine* percebe a aproximação do jogador em certas áreas do cenário e faz o carregamento daquela área em segundo plano, de modo que já estará carregada quando o jogador passar por ela. Esta característica foi a que permitiu que alguns jogos fossem jogados do início ao fim com único carregamento. A RAGE também possui o recurso de mudança temporal bem avançado, sendo que ela se encarrega de mudar a luz e o clima do jogo com o passar dos dias e dos meses. Mais sobre a RAGE pode ser encontrado em (RAGE, 2010)

KODU Game Lab

Esta *game engine* é na verdade um jogo para fazer jogos. A interface e a linguagem de programação totalmente visual e intuitiva foram projetadas para serem acessíveis para crianças (KODU, 2009). Com suporte para PC e Xbox 360, a Kodu, apesar de muito simples não é especializada em gêneros específicos de jogos e demonstra o quão fácil a tarefa de desenvolvimento de jogos pode se tornar. A figura 2.3 mostra algumas imagens do desenvolvimento de jogos com a KODU.

Figura 2.3 – Game engine KODU

2.2 Sistemas de autoria 3D

A animação 3D se tornou uma tecnologia fundamental em aplicações como visualização, simulação e sistemas de treinamento, visto que os benefícios da imersão 3D são

inegáveis (EBERLY, 2000). A necessidade de criação de cenas e animações 3D cresceu nas mais diversas áreas como medicina, indústria e construção civil. Desta forma viu-se a necessidade de criação das animações de forma fácil, rápida, simples, barata e ainda sem conhecimento profissional na área de simulação ou animação 3D. Para suprir estas necessidades, existem as ferramentas de autoria 3D.

Para ilustrar a importância destas ferramentas, pode-se utilizar o exemplo de uma fábrica de motor de carro: um funcionário recém contratado para o departamento de manutenção precisa se familiarizar com as peças e funcionamento de um modelo de motor. Porém, o difícil manuseio das peças reais e a impossível visualização do funcionamento interno deste motor impossibilitam que o funcionário realize o treinamento. Para resolver este problema, outros funcionários da área de manutenção que já estavam familiarizados com o motor, criaram uma animação 3D, em uma ferramenta de autoria, com as peças do motor em funcionamento. O funcionário recém contratado agora pode conhecer as peças do motor e saber como elas agem. O melhor disto é que os criadores da animação não precisaram de conhecimentos de programação ou mesmo de animação profissional. Precisaram apenas conhecer a ferramenta de autoria e o processo que estavam criando, no caso do exemplo, o funcionamento do motor.

2.2.1 Game engine versus Sistema de autoria 3D

A comunidade *Intelligent Tutoring System*, popularizou o termo sistema de autoria, referindo-se a um sistema de computador que permite um grupo não-restrito ou não específico criar outro sistema de computação, ou seja, um sistema de autoria permite que um não programador crie um sistema com características comuns de um software sem utilizar conhecimentos de programação por linguagens de computador convencionais (ONG e RAMACHANDRAN, 2000). A grande semelhança com as características de uma *game engine* é notória, porém o termo sistema de autoria 3D é usado quando o principal produto do sistema de autoria pode ser produzido com o uso de linguagem visual (EVENS e MICHAEL, 2005). Segundo (EBERLY, 2000) esta semelhança se deve a herança recebida pelas *game engine's*, as quais sempre foram as vanguardas na área de autoria 3D. Em contrapartida, uma *game engine* não será um sistema de autoria enquanto necessitar fundamentalmente de conhecimentos de programação. Em muitas *game engine*'s o que se vê é o uso de

programação por *scripts*¸ uma linguagem interpretada criada especialmente para resolução de um determinado problema, o que aumenta o nível de abstração sem perder a liberdade de programação. Mas ainda assim não possui a simplicidade do uso de uma linguagem totalmente visual, apreciada nos sistemas de autoria.

Existe ainda a chamada *engine* gráfica, que é um módulo de software, podendo ser uma ferramenta pronta ou um conjunto de bibliotecas de programação, com propósito de resolver problemas de renderização gráfica. Por exemplo: a *engine* Ogre se propõe a resolver o problema de renderização 3D; A *engine JGame* é uma biblioteca gráfica 2D para Java.

Em resumo tem-se:

- *Engine* gráfica cuida apenas da renderização gráfica da cena;
- *Game Engine* cuida de todas as abstrações para criação de um jogo, seja gráfico, som, IA ou rede, podendo ou não ser totalmente servido de linguagem visual;
- Sistema de autoria 3D cuida de todas as abstrações da produção a que se propõe, assim como uma *game engine*, porém um sistema de autoria 3D é dotado de linguagem visual para o desenvolvimento.

2.2.2 Exemplos de sistemas de autoria 3D

Grande parte dos sistemas atuais trabalha na área de criação de modelos 3D, a exemplo de 3D Studio Max ou Maya. Outros sistemas são modeladores de terreno e cenários 3D. Porém, esta dissertação, irá detalhar somente os sistemas de autoria 3D que permitem a criação de animações em tempo real com interação do usuário.

Anark Media Studio

A plataforma *Anark Media* (ANARK, 2010) é um avançado sistema de desenvolvimento de aplicações 3D, podendo ser usado para a autoria de simulações, treinamento ou simples visualização. Seu ponto forte é a grande capacidade de interação do usuário com a animação criada, o que é fundamental para aplicações de treinamento. Este sistema de autoria é totalmente livre de conhecimentos de programação (apesar de possuir

suporte para scripts) ou técnicas avançadas de animação. A figura 2.4 mostra uma imagem do sistema de autoria da plataforma Anark Media.

Figura 2.4 - Interface do sistema de autoria Anark Media Studio

O produto criado no *Anark* necessita de um programa cliente ou plugin para navegador, distribuído pela empresa, para ser executado, ou pode ser distribuída em formato de vídeo, perdendo assim os recursos de interatividade. Este sistema possui ainda um sistema de colisão simples, porém sem qualquer simulação física. Abaixo seguem algumas características deste sistema de autoria 3D.

- Visualização de vídeo no ambiente 3D;
- Múltiplas câmeras;
- Implementação de LoD (Level of Detail) em objetos 3D e texturas;
- Visualização de texto formatável;
- Criação de animações auxiliadas por Linha-de-tempo;
- Conexão com banco de dados para carregamento de recursos;
- Manual com texto e tutoriais em vídeo;

- Necessita de uma placa de vídeo NVIDIA Quadro FX Series 1700.

Unity 3D Editor

Este sistema de autoria é muito usado também como *game engine* para jogos infantis e educativos. A grande versatilidade deste sistema, combinando facilidades de funções como “clique-e-arraste” com o uso de scripts, torna-o muito eficiente no quesito tempo de produção (UNITY, 2010). Este sistema também é utilizado para a visualização de obras arquitetônicas, onde o grande diferencial está na simulação de navegação de um personagem avatar por dentro do cenário criado. Este sistema é um grande exemplo da equivalência existentente entre *game engine* e sistemas de autoria 3D, onde neste caso nenhuma diferença pode ser percebida a não ser no produto criado. A figura 2.5 mostra imagens de aplicações desenvolvidas com o *Unity 3D Editor*.
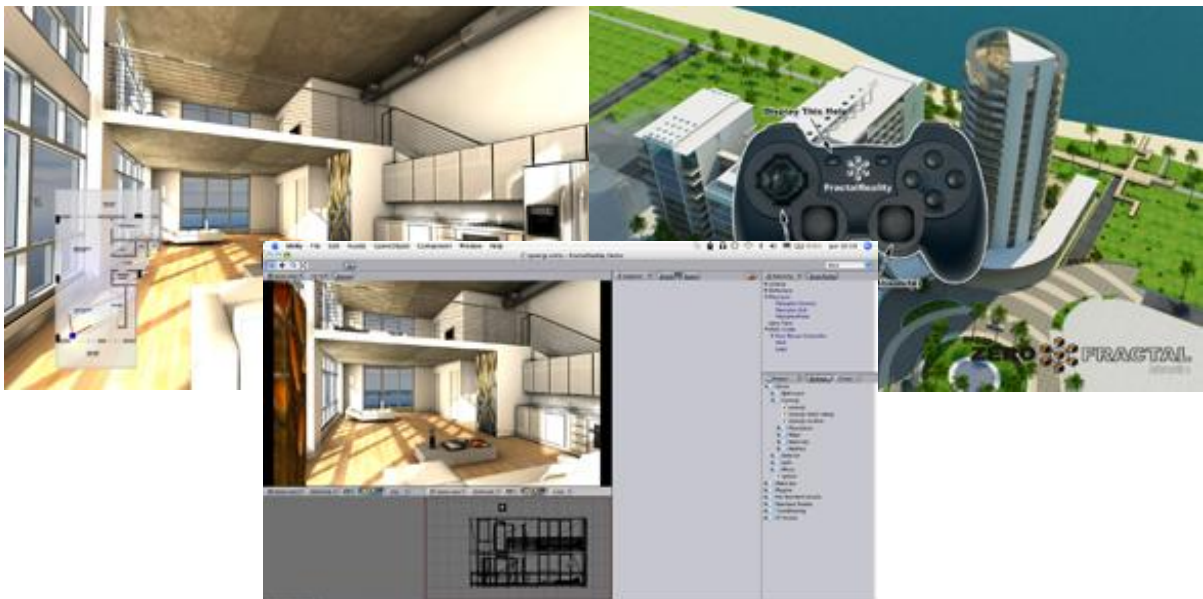
Figura 2.5 – Sistema de autoria Unity 3D em uso

Podem-se destacar ainda algumas características do *Unity:*

- Multiplataforma (PC, Mac, Nintendo Wii, Apple IPhone e Web);
- Multi-Gênero, a exemplo de simulações, quebra-cabeça, RPG, corrida e jogos educativos;

- Criação de modelos de desenvolvimento que podem ser usados em diferentes projetos ao mesmo tempo;
- Permite o uso de diferentes linguagens para script no mesmo projeto;
- Distribuição e visualização das animações via WEB.

Thinking Worlds

Este sistema de autoria 3D possui dois grandes diferenciais: é totalmente livre de scripts, podendo tudo ser feito através da interface intuitiva da ferramenta; as simulações criadas nele podem ser distribuídas para serem executadas em navegadores compatíveis com a tecnologia Flash. Suas principais aplicações estão na área de treinamento. A figura 2.6 mostra a interface do *Thinking Worlds* (WORLDS, 2009).
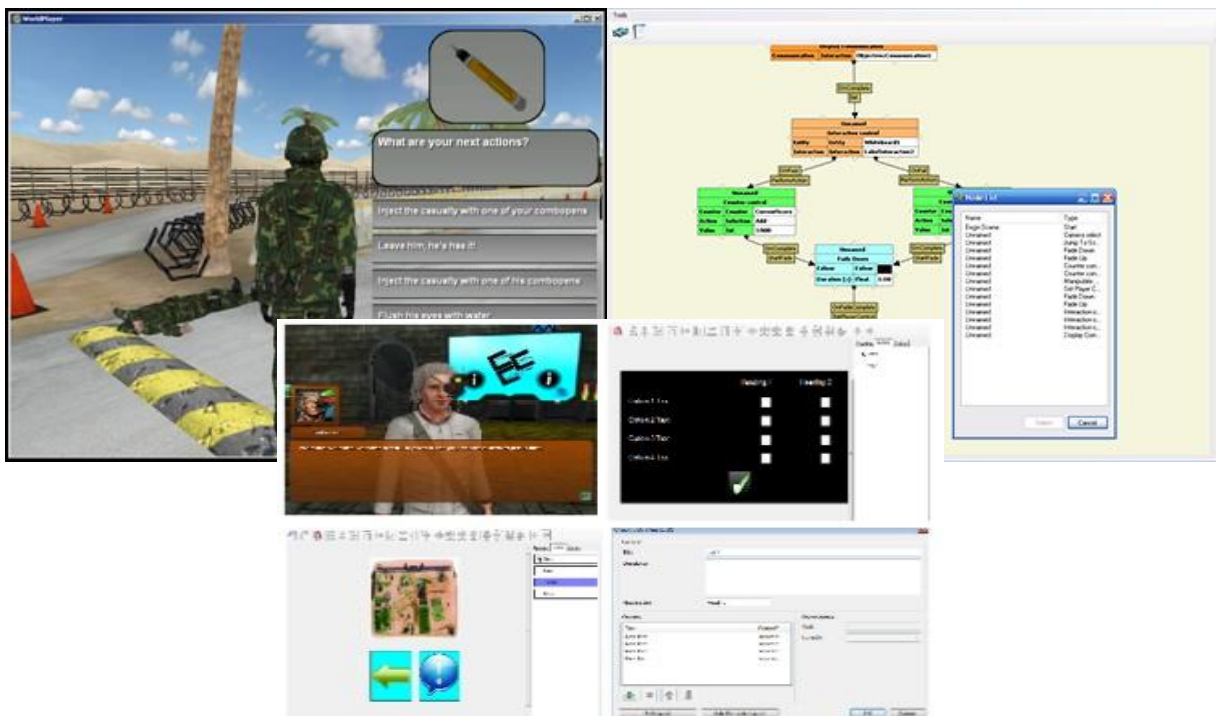
Figura 2.6 - Interface do Thinking Worlds

O sistema de autoria Thinking Worlds foi projetado para criar simulações e treinamentos em 3D, porém pode ser usado para criar jogos, percebendo-se certa limitação

para isso com relação a outras *game engine's*. Desta forma o Thinking Worlds apresenta uma característica desejada na área de treinamento, que é a exportação no padrão SCORM (conjunto de protocolos para aprendizado via web, (SERVICES, 2001)).

NGRAIN Producer

O sistema de autoria NGRAIN é especializado em criação de animações de procedimentos e criação de tarefas de treinamento e simulação em tempo-real. Semelhantemente a outros sistemas de autoria, NGRAIN é um sistema que não necessita de conhecimentos de programação e, segundo o site da empresa (NGRAIN, 2010), pode ser aprendido em somente dois dias. O sistema utiliza uma interface dotada de linha-do-tempo para manejar as animações criadas, que vão desde rotações simples a narrações com som. O NGRAIN permite também a criação de tarefas para a construção de treinamentos. A figura 2.7 apresenta a interface no momento da construção de uma simulação.
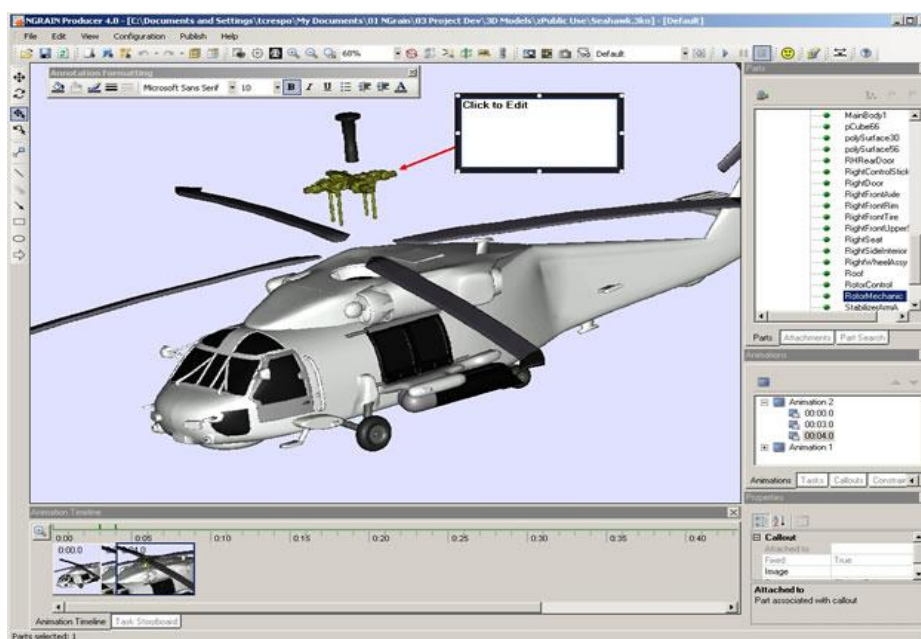
Figura 2.7 - Interface do NGRAIN

O NGRAIN é dividido em partes com funções bem definidas para facilitar a criação, distribuição e utilização dos projetos desenvolvidos:

- Producer – É o sistema de autoria, propriamente dito. Nele podem-se criar animações, importar recursos e criar tarefas de treinamento. O *Producer* cria projetos em conformidade com o padrão SCORM;
- Ferramentas de conversão – Ferramentas que fazem a distribuição do projeto NGRAIN para os mais diversificados formatos, entre eles Microsoft PowerPoint, PDF e executável Microsoft Windows;
- Exploração – Permite navegar e controlar todo o cenário criado no *Producer*. Com a ferramenta de exploração pode-se mover ou esconder peças para melhor visualização, navegar com a câmera, retirar fotos, fazer anotações, etc;
- Visualização – Esta ferramenta é a que mostra a simulação ou treinamento da forma como acontece na prática e como foi concebida pelo projetista no *Producer*. Nesta ferramenta o usuário não possui interação com o ambiente virtual, podendo somente pausar ou continuar a animação;
- Prática – O usuário da simulação pode interagir com ela realizando as tarefas necessárias para concluir o treinamento;
- Teste – É semelhante à ferramenta de prática, porém no teste o usuário estará sendo supervisionado e avaliado quanto a erros e acertos no decorrer do procedimento simulado.

2.3 UGV – Unidade Geradora Virtual

O sistema UGV, desenvolvido pelo Laboratório de Realidade Virtual da UFPA (LARV, 2010), é um software criado para visualização e treinamento de processos de manutenção e operação em uma usina hidrogeradora através do uso de técnicas de RV com gráficos 3D. UGV tinha como objetivo principal demonstrar sete processos ocorridos cotidianamente na UHG . Um estudo mais aprofundado da UGV pode ser encontrado em (PAMPLONA, 2006), (VILHENA e FILHO, 2007) e (SOUSA, 2009). A figura 2.8 mostra uma imagem do software UGV em uso.
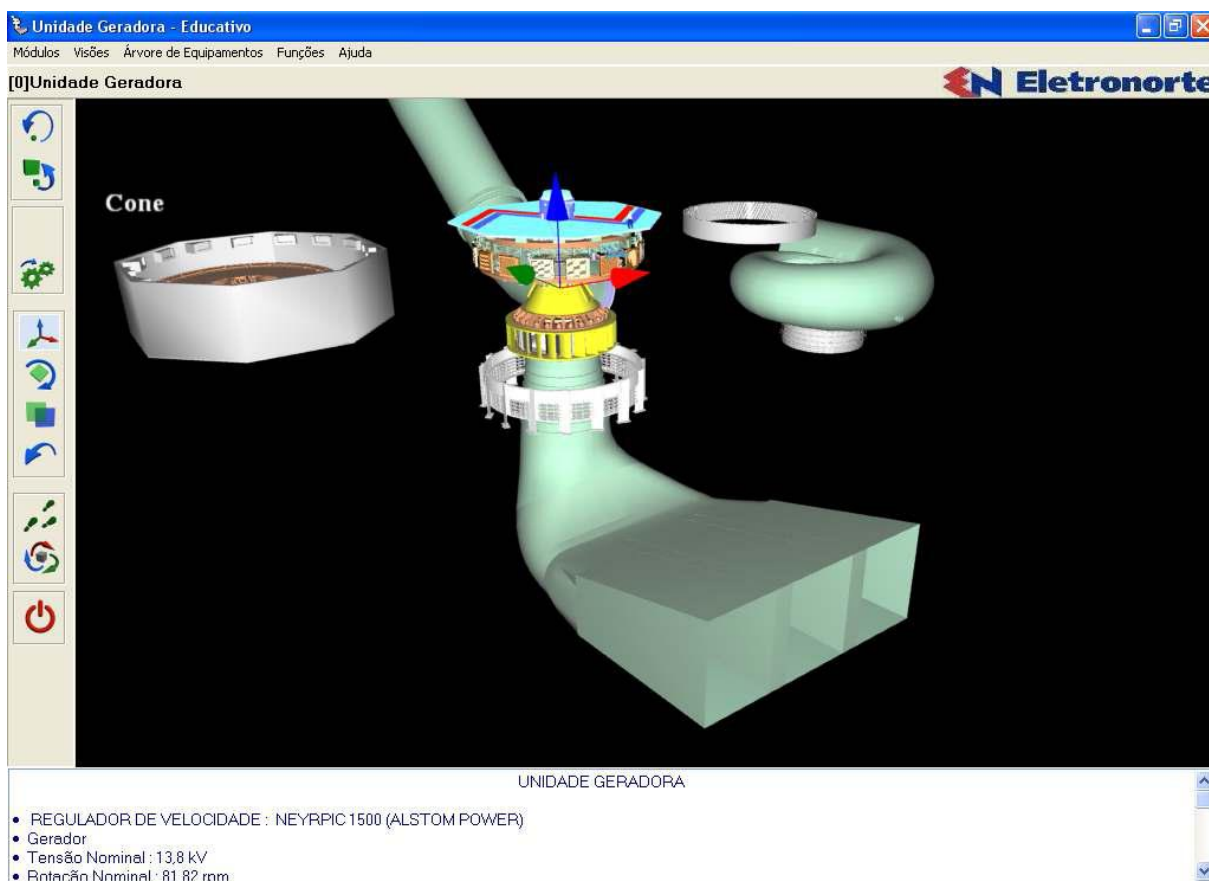
Figura 2.8 - Software UGV

2.3.1 Características e funcionalidades

O software UGV é dividido em módulos. Esta divisão permite o uso do software por diferentes perfis de usuários. Além disso, qualquer um dos módulos pode ser carregado em qualquer momento da execução do software. Os principais módulos da UGV são:

- Módulo educativo: permite que o usuário visualize os modelos 3D que representam peças de uma unidade hidrogeradora. Neste módulo existe a liberdade de navegar pelo ambiente, rotacionar e mover objetos, além de visualizar informações reais das peças representadas nos modelos 3D;
- Módulos de operação de partida e parada: nestes módulos, o usuário pode visualizar animações de dois processos existentes em uma usina: a partida (alcançar o estado de geração de energia) e a parada (alcançar o estado em que não há geração de energia) de uma turbina hidrogeradora. O usuário não possui interação com o ambiente virtual, podendo somente pausar ou continuar as animações;

- Módulo de operação de contingencias: Neste módulo é possível verificar uma simulação de distúrbios gerados na usina hidrelétrica e verificar a conseqüência desses distúrbios em uma unidade da usina, tudo isto claro, usando recursos de realidade virtual. Diferentemente dos outros módulos de operação, neste módulo não existe uma animação sem iteração, mas sim uma animação baseada em simulações matemáticas, podendo ser verificado o movimento do conjunto distribuidor da máquina no ambiente virtual de acordo com ações do usuário. Aqui é possível navegar livremente no ambiente, porém não é possível manipular as peças como no módulo educativo;
- Módulos de manutenção: No módulo de manutenção, o usuário toma o papel de um mantenedor da unidade geradora, simulando a troca de peças específicas no próprio ambiente virtual. A simulação é assistida por um processo inteligente do programa, de modo que o usuário seja avaliado por seu desempenho na manutenção da unidade e seja emitido e salvo um relatório para controle do treinamento, tudo gerenciado por um banco de dados simples com informações de usuário. A UGV original possui quatro tipos de manutenção, cada uma em diferentes partes da usina. Este módulo pode ser utilizado de três maneiras diferentes pelo usuário, o qual pode apenas visualizar a animação da manutenção, pode realizar a manutenção de maneira assistida com instruções de como proceder, ou ainda realizar a manutenção sem assistência, porém com supervisão do rendimento. Nas duas últimas maneiras, o usuário interage com o ambiente virtual manipulando os modelos 3D como se estivesse na manutenção real.

É importante saber que a UGV possui os módulos já desenvolvidos, portanto qualquer modificação nas animações ou simulações do software não poderia ser feita sem que o código fosse reprogramado. Apesar disto, muitas funções interessantes foram apresentadas pela UGV, podendo-se destacar a árvore de peças com a lista completa dos objetos 3D do módulo em execução, supervisão autônoma das simulações de treinamento, manipulação dos objetos 3D, incluindo translação, rotação e transparência e informação textual das peças.

2.3.2 Arquitetura computacional

UGV foi implementado usando o ambiente de desenvolvimento Delphi (BORLAND, 2010), sendo este orientado a objetos e o ambiente tridimensional que representa a usina hidrogeradora foi construído utilizando-se a biblioteca open source GLScene (DEGIOVANI,

2003) para Delphi a qual é uma biblioteca 3D baseada em OpenGL, fornecendo componentes visuais e objetos, permitindo a descrição (grafo de cena) e renderização das cenas 3D. Os modelos 3D utilizados para composição da cena foram criados na ferramenta 3D Studio Max (AUTODESK, 2010). Utilizaram-se ainda arquivos no formato XML para armazenamento de informações como arranjo das peças na cena 3D e informações de usuários do programa.

Existe um componente de carregamento recebendo as solicitações do usuário para alocar em memória os modelos 3D. Esse mesmo módulo também carrega os arquivos de usuários e de peças, que estão armazenados em arquivos XML. Os arquivos de usuários contêm descrições como: nome de usuário, senha, avaliação, status da tarefa, e os arquivos de peças com descrições dos modelos das peças como: nome da peça, dados físicos, posicionamento no ambiente, entre outros. Há o agente de supervisão responsável pela administração dos dados referentes do treinando, registrando informações pessoais como nome e senha, e também informações de avaliação do módulo de treinamento, tais como, desempenho do treinando e o status de conclusão dos procedimentos de manutenção. A interface promove a interação entre o sistema e o usuário através da renderização gráfica da cena 3D e os elementos de interação convencionais como botões, caixas de texto e painéis.

3 DESENVOLVIMENTO DO SISTEMA DE AUTORIA ITV

Uma instrução técnica é um conjunto de procedimentos ou normas que devem ser realizados fielmente a fim de resolver uma tarefa ou se enquadrar em um padrão (NETTO e TAVARES, 2006). As IT’s (Instruções Técnicas) são de grande importância na indústria, pois, mostra um caminho otimizado que resolva um problema, de modo que não haja acidentes, seja rápido e utilize o mínimo de recursos disponíveis. IT’s são utilizadas em inúmeras áreas além da indústria como no Corpo de Bombeiros, piscicultura ou lanchonetes.

Para exemplificar, a Eletronorte S.A. possui um procedimento de desmontagem da caixa de vedação da turbina hidrogeradora. Este procedimento possui uma Instrução Técnica definindo desde parâmetros de segurança até os materiais utilizados na limpeza das peças. Esta IT é totalmente textual e cita nomes de peças a serem desmontadas sem sequer descrevê-las ou apresentar fotos da mesma. Logo, a execução desta IT não seria realizada facilmente por alguém que não conhece o procedimento na prática.

Uma Instrução Técnica Virtual é a transposição de todo o cenário descrito na IT textual para o mundo virtual em 3D. A idéia central das ITV’s é transformar a descrição textual de uma IT textual comum em uma demonstração visual e interativa do processo, tornando o aprendizado mais simples e intuitivo. Vale ressaltar que as ITV’s não substituem IT’s textuais, porém elas servem como auxílio no treinamento e ambientação.

Em 2007, o software UGV apresentou sete módulos que simulavam instruções técnicas de operação e manutenção da Usina Hidrelétrica de Tucuruí. Como dito anteriormente na seção 2.3, estas instruções não podiam ser modificadas, ou não se podia fazer novas instruções, a menos que houvesse recodificação. Em (SOUSA, 2009) pode-se analisar a importância do uso de ferramentas de realidade virtual na engenharia e indústria, além de validar a importância de UGV no treinamento de profissionais e alunos da área de geração de energia. Sabendo-se então da grande eficácia de simulações e treinamentos em realidade virtual um novo projeto foi proposto no início de 2008 pela Eletronorte S.A.. Com o objetivo de virtualizar 10 instruções técnicas em 24 meses, porém este novo projeto tem um objetivo maior: desenvolver um sistema de autoria que possa criar as Intruções Técnicas Virtuais de forma rápida e sem recodificação, diferentemente do que ocorreu com UGV.

Este capítulo mostra o projeto do sistema de autoria desenvolvido de 2008 a 2010, chamado de Sistema de Autoria ITV, destacando os requisitos do sistema, as ferramentas utilizadas no desenvolvimento e os conceitos do funcionamento do software.

3.1 Requisitos

Requisitos, em engenharia de software, são objetivos ou restrições estabelecidas por clientes e usuários que definem as diversas características do sistema. Um conjunto de requisitos pode ser definido como uma condição ou capacidade necessária que o software deve possuir para que o usuário possa resolver um problema ou atingir um objetivo ou para atender as necessidades ou restrições da organização ou dos outros componentes do sistema.

Tradicionalmente, os requisitos de software são separados em requisitos funcionais e não-funcionais (ROGER, 2006). Os requisitos funcionais são a descrição das diversas funções que clientes e usuários querem ou precisam que o software ofereça. Eles definem a funcionalidade desejada do software. A especificação de um requisito funcional deve determinar o que se espera que o software faça, sem a preocupação de como ele faz. Requisitos não-funcionais são as qualidades globais de um software, como usabilidade, desempenho ou custos.

Os requisitos para o Sistema de Autoria ITV foram definidos de acordo com os usuários finais do software, a saber, os funcionários da Eletronorte S.A.. Dentre os requisitos funcionais do sistema de autoria estão:

- O sistema deve possuir um modo de execução que permita a criação e edição de ITV's, que é o próprio sistema de autoria;
- O sistema deve possuir um modo de execução que permita a visualização e interação do usuário sobre as animações, simulações e treinamentos criados. Este modo de execução não deve permitir a edição ou criação de ITV's. Este modo de execução do Sistema de Autoria ITV é denominado módulo de usuário;
- O Sistema de Autoria ITV deve ser totalmente visual, não apresentando programação por linguagem ou *script.*

Os principais requisitos não- funcionais são:

- O tempo de desenvolvimento será de no máximo 24 meses;

- O sistema deverá ter um tempo de resposta aceitável para aplicações com interação;
- O software de ser funcional em plataforma Windows;
- O software deve ser acessível a usuários não familiarizados com softwares semelhantes.

Todos estes requisitos foram alcançados e serão demonstrados com detalhes no capítulo 4, implementação do sistema de autoria.

3.2 Ferramentas utilizadas

O projeto do sistema de autoria fez uso de API's (*Application Programming Interface*) livres para linguagem de programação C++. Cada API utilizada ficou encarregada de uma ou mais partes do software, por exemplo, renderização gráfica, matemática, colisão, som, etc. Este tópico apresenta as ferramentas usadas na implementação do Sistema de Autoria ITV e a sua importância para o software.

3.2.1 Linguagem C++

A linguagem de programação C++ é uma das linguagens de alto nível mais utilizadas no mundo, seja comercialmente ou não. A sua popularidade é percebida principalmente no meio científico, devido ao seu desempenho e facilidade de utilização. É uma linguagem orientada a objeto e faz uso de quase todos os conceitos deste paradigma, como herança e polimorfismo (DEITEL, 2006). É muito respeitada pela liberdade que dá sobre a máquina, característica que também provoca repúdio entre alguns. A escolha de C++ para ser a linguagem de programação deste projeto não foi em vão: C++ é a linguagem preferida pelos programadores de jogos e aplicações gráficas 3D (DEGOES, 1999).

Toda a codificação do Sistema de Autoria ITV foi feita em alto-nível na linguagem C++ e em nenhum momento foi necessário o uso de scripts, linguagem de máquina, ou codificação por macros. O compilador C++ utilizado foi o Microsoft Visual C++.

3.2.2 OGRE 3D - Object-Oriented Graphics Rendering Engine

OGRE (Object-Oriented Graphics Rendering Engine) é uma engine 3D escrita em C++ desenvolvida com o intuito de tornar a produção de aplicações gráficas 3D mais fácil e

rápida, através do uso de suas dezenas de classes prontas. A biblioteca está em constante atualização e recebe ajuda de milhares de desenvolvedores do mundo inteiro, já que OGRE tem código aberto, aliás, a OGRE, é uma das bibliotecas gráficas 3D de código-livre mais utilizadas no mundo, segundo (JUNKER, 2006). Entre outras características, a biblioteca é multiplataforma, funcionando em plataforma Windows, Linux e Mac, o que leva a zero o trabalho de portagem de código entre plataformas, caso seja usado OGRE puro. Devido a essa característica, a OGRE utiliza duas engines 3D de baixo nível: OpenGL, usada em Windows, Linux e Mac e Direct3D usada em Windows.

Esta biblioteca foi escolhida para a implementação do Sistema de Autoria ITV porque é uma biblioteca já validada pela grande comunidade de desenvolvedores de jogos e aplicações 3D, além de apresentar características que são importantes para a estrutura desejada no projeto, como arquitetura flexível (OpenGL e Direct3D) e uso de scripts para definição de parâmetros nos modelos 3D. Por ser somente uma *graphic engine* (*engine* que trata somente da renderização gráfica) e não uma *game engine*, OGRE não possui gerenciamento de física, som ou mesmo interface, porém existem inúmeras adições à biblioteca, desenvolvidas pela imensa comunidade do OGRE, que resolvem muito bem estas tarefas. O uso da biblioteca é bastante facilitado por manuais, documentação, tutoriais e fóruns disponíveis na Internet, sendo mais uma vantagem da biblioteca. A figura 3.1 mostra algumas aplicações que utilizam OGRE.

Figura 3.1 - Jogos feitos com a biblioteca OGRE 3D

Abaixo estão listadas as características técnicas que fizeram OGRE ser a escolha para o desenvolvimento do Sistema de Autoria ITV.

- Uso simples da interface orientada a objeto, arquitetada de modo a diminuir o esforço na criação de cenas 3D;
- Independente do conhecimento de linguagens 3D de baixo nível como Direct3D e OpenGL;
- Requisitos comuns como verificação de profundidade e gerenciamento de transparência são feitos automaticamente, salvando muito tempo de desenvolvimento;
- Estrutura de classes limpa e totalmente documentada;
- Uma engine estável comprovada por muitas aplicações comerciais;
- Suporte a Direct3D e OpenGL;
- Suporte a Windows (todas as principais versões), Linux e Mac OSX;
- Poderosa linguagem de declaração de material, podendo ser usada como script;

- Suporta instruções avançadas de sombreamento como Vertex e Pixel Shader, tanto em linguagem de baixo nível (assembler), como de alto nível, com suporte a parâmetros pré-definidos como posição de luz ou posição de câmera;
- Possibilidade de definição de várias técnicas em um mesmo material, facilitando a modificação dinâmica deste;
- Múltiplos níveis de detalhes em materiais, tornando a aplicação mais leve;
- Carregamento nativo de texturas nos formatos PNG, JPEG, TGA, BMP ou DDS, incluindo formatos menos populares como texturas 1D, texturas volumétricas e texturas comprimidas (DXT/S3CT);
- Flexível formato de geometria, fazendo separação de vértices, grupo de vértices e faces;
- Uso de suavização automática para superfícies curvas, otimizando o nível de detalhe da geometria sem alto custo de processamento;
- Nível de detalhe em geometria, automático ou manual;
- Flexível e customizável gerenciador de cena, facilitando a criação de arquiteturas em aplicações robustas;
- Tipos especializados de cena, como cena *indoor* ou cena *outdoor*, melhoram o rendimento computacional em aplicações específicas;
- Sistema de partículas podem ser criados por script, além de utilizarem técnicas de otimização de desempenho;
- Objetos transparentes são gerenciados automaticamente, evitando o problema de ordem errada de renderização;
- Suporte a recursos populares de arquivamento, como ZIP e PK3;
- Gerenciamento de memória pode ser rastreado e customizado pelo programador para um desenvolvimento e execução mais seguro;
- Facilidade de utilização de OGRE com outras bibliotecas de software.

3.2.3 Qt

Qt é um conjunto de aplicações e bibliotecas multiplataforma, desenvolvido pela empresa Nokia, para o desenvolvimento de programas com interface gráfica. Seu conjunto de bibliotecas é desenvolvido em C++ e para C++ e possui mais de 400 classes, que vão desde

interfaces GUI's (*Graphics User Interface*) até redes, tudo com documentação completa, além de demonstrações, exemplos, tutorias e manuais.

O código em Qt é multiplataforma, suportando de Windows 98 até o Seven, Mac OSX, Linux, Solaris, HP-UX, IRIX, AIX, BSD, Tru64 UNIX, além de poder ser usada em sistemas embarcados e sistemas móveis. A figura 3.2 mostra algumas aplicações desenvolvidas com Qt.
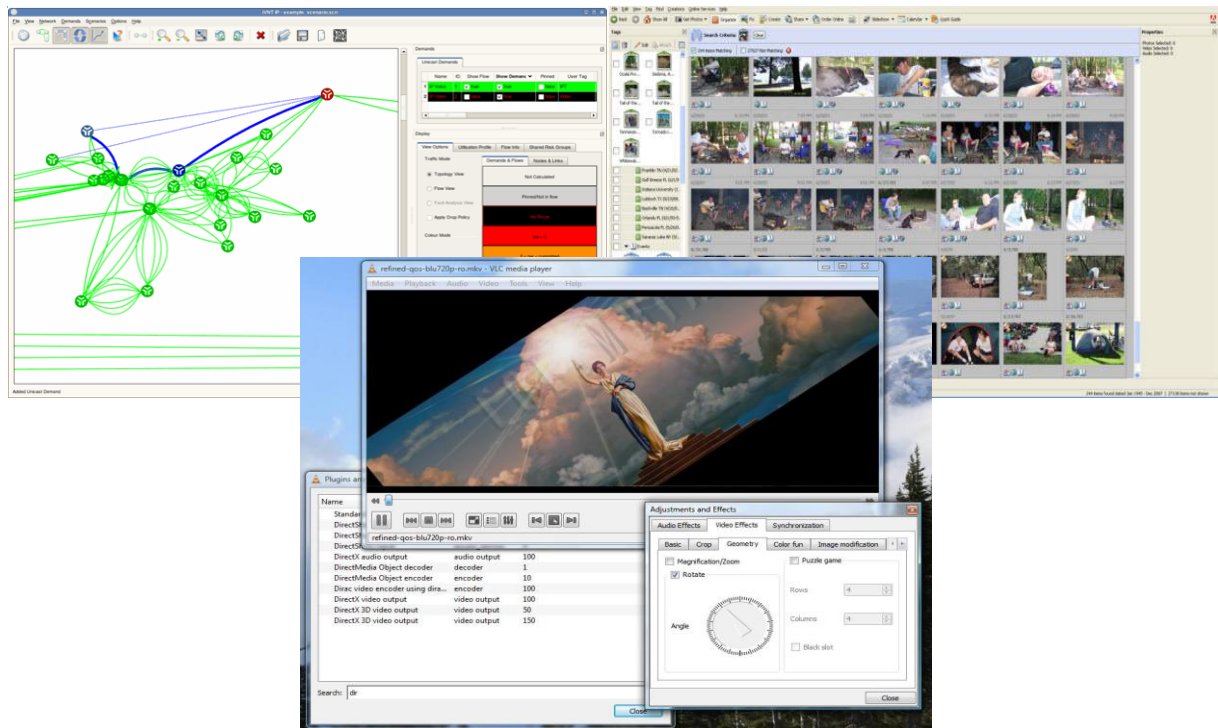
Figura 3.2 - Aplicações utilizando Qt

O Sistema de Autoria ITV necessitava de uma GUI nativa exterior ao ambiente virtual (a interface será detalhada no capítulo 4), logo se verificou que seria necessário uma biblioteca que permitisse um rápido aprendizado e facilidade de utilização, características encontradas na biblioteca Qt (BLANCHETTE e SUMMERFIELD, 2008). Dentre algumas vantagens do uso de Qt no desenvolvimento do Sistema de Autoria ITV estão:

- Código fonte aberto;
- Utilização Gratuita;
- Ferramentas de desenvolvimento que aumentam a produtividade: Qt Designer (Modelamento da GUI e layout), Qt Linguist (Tradução), Qt Assistant

(Documentação), QMake (transparência na construção de sistemas multiplataformas);

- Integração com a IDE Microsoft Visual C++ com funções úteis em tempo de desenvolvimento como complementação de código e sinalização de sintaxe.
- Qt Designer embutido na IDE permitindo um desenvolvimento de interfaces do tipo clique e arraste;
- Comunidade extensa com centenas de desenvolvedores pelo mundo, dentre alguns exemplos comerciais feitos com Qt estão Adobe Photoshop Elements e Google Earth.

3.2.4 OpCoDe - Optimized Collision Detection

O Sistema de Autoria ITV possui uma ferramenta de seleção das peças através do mouse, ferramenta comumente chamado de *picking*, um tipo de detecção de colisão, (BERGEN, 2003). A biblioteca OGRE já possui algumas classes para implementar o *picking*, porém a seleção não é precisa, isto porque ela é feita baseada no *bounding box* da peça, que é uma caixa que envolve o objeto e é geralmente usada para detecção de colisão, veja a figura 3.3.
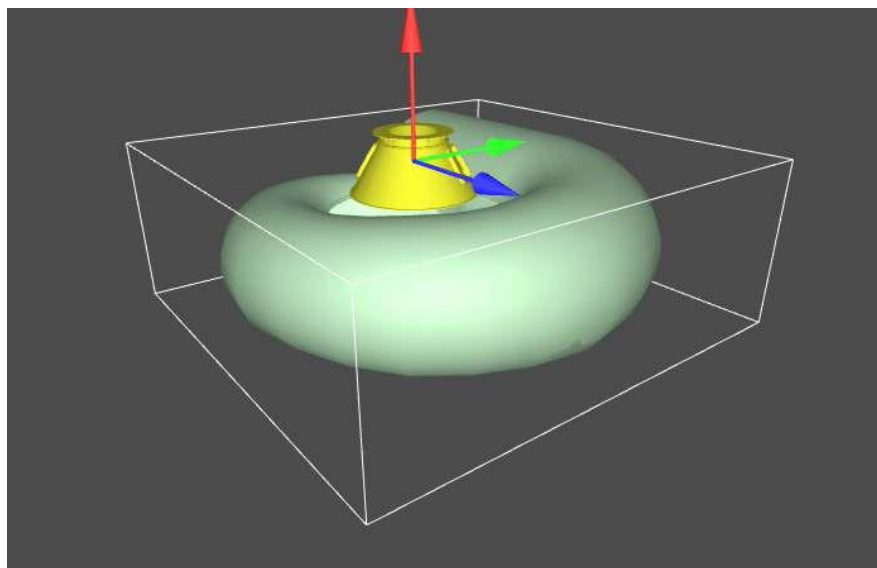
Figura 3.3 - Representação do bounding box

Esta imprecisão não é boa para o sistema de seleção da ITV, já que o usuário poderia clicar em uma peça, mas selecionar outra equivocadamente. Portanto o ideal é que a seleção seja feita com o clique exato em qualquer parte da peça, ou seja, somente clicando em uma das faces da geometria do objeto. Para isso foi utilizada a biblioteca para detecção de colisão OpCoDe (Optimized Collision Detection), que integrada com OGRE chama-se OGREOpCoDe. Esta é uma biblioteca de código aberto e dá suporte somente a detecção da colisão, deixando por conta do desenvolvedor o tratamento físico devido. OpCoDe é usado como base em outros gerenciadores de física como o ODE. Para saber mais sobre OpCoDe consulte (OPCODE, 2008).

3.2.5 FMOD API

FMOD é uma biblioteca para reprodução e criação de áudio desenvolvida pela *Firelight Technologies*, possuindo suporte a muitos formatos de áudio em diversas plataformas. Suas principais vantagens estão no custo, sendo gratuito para projetos não-comerciais, e a popularidade, o que torna a busca e aprendizado por uma solução muito mais rápidos. Algumas aplicações que utilizam o FMOD são: Unreal Engine 3, Unity engine, Forza Motor Sport 3 e a série de jogos Guitar Hero (FMOD, 2010). O Sistema de Autoria ITV utiliza a biblioteca em alto nível FMOD somente para reprodução de sons, porém a reprodução pode se dar em stereo ou em três dimensões.

3.2.6 XML - eXtensible Markup Language

O XML (eXtensible Markup Language) é uma linguagem de marcação para documentos estruturados. Essa estrutura de um documento XML possui informações tanto de conteúdo, como do comportamento do conteúdo, por exemplo, O XML pode guardar uma foto e também informações sobre o que fazer com esta foto, como girar ou dar zoom. Uma linguagem de marcação é uma forma de identificar estruturas em um documento, e o XML vai além, ele deixa o desenvolvedor livre para criar suas próprias marcações e assim customizar uma estrutura (DYKES e TITTEL, 2005). As marcações do XML são chamadas de *tags* e podem ser dispostas em níveis hierárquicos.

A UGV descrita na seção 2.3 já utilizava com sucesso documentos XML, funcionando como um banco de dados, guardando informações de nome de peças, posicionamento e descrição. Desta vez, o Sistema de Autoria ITV utiliza o XML para os mesmos objetivos, porém guardando mais informações sobre uma peça: nome para uso interno, rótulo com formatação, posição, descrição formatada de objetos, fotos relacionadas e outras informações internas, além de descrever um arquivo de simulação ITV, como será detalhado no capitulo 4. Desta forma o software não precisa ser recompilado para o ajuste das informações de peças e animações.

3.3 Arquitetura do Sistema de Autoria ITV

O Sistema de Autoria ITV é composto de módulos percebidos apenas pelo ponto de vista do desenvolvedor. Cada módulo possui funções bem definidas para resolução de algumas tarefas totalmente transparentes ao usuário do software. Por exemplo, o módulo de carregamento de recursos é extremamente importante para o gerenciamento eficiente da memória, porém o usuário do software desconhece este gerenciamento, vendo somente o objeto já carregado no ambiente virtual. A figura 3.4 mostra os módulos componentes do Sistema de Autoria ITV e suas relações
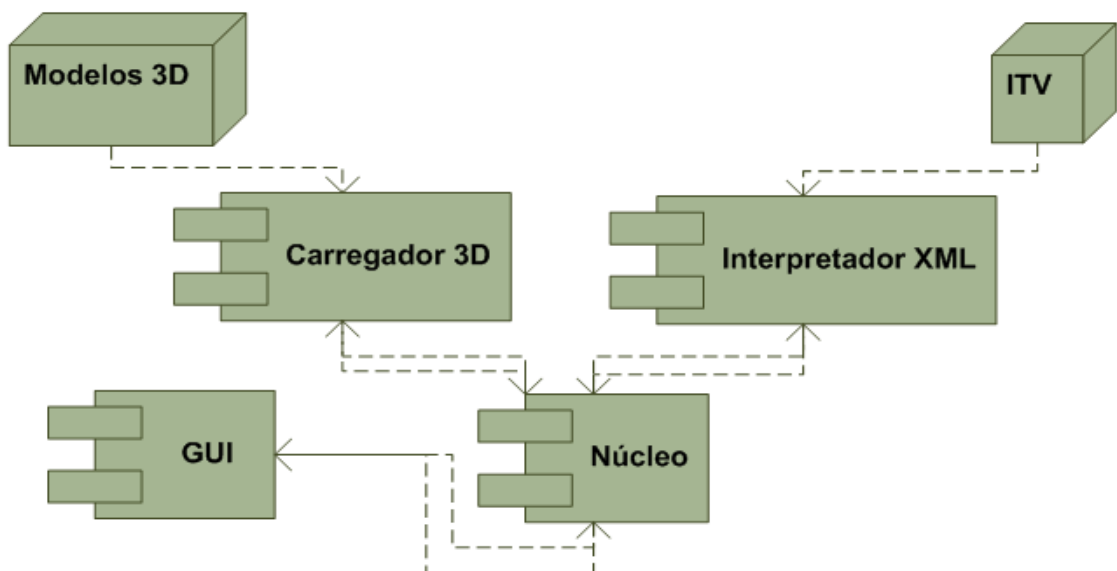
Figura 3.4 - Arquitetura do Sistema de Autoria ITV

3.3.1 Núcleo

O Núcleo é o módulo mais importante do Sistema de Autoria ITV. É responsável pelo controle da cena 3D e de toda a construção e exibição da simulação, incluindo som, renderização 3D e supervisão de treinamentos. O Núcleo controla ainda as funções disponíveis no módulo Interpretador XML. A figura 3.5 representa os sub-módulos constituintes do núcleo da ITV.

Objetos

São os elementos que podem ser manipulados pelo usuário recebendo algum tipo de animação que vai compor a simulação final. Os objetos do Sistema de Autoria ITV são câmera, modelos 3D, partículas, sons e painéis textuais. As funções e a utilização de cada objeto serão explicadas no capítulo 4.

Animações

Cada objeto pode receber algum tipo de animação, por exemplo, uma partícula pode ser movida de um ponto a outro em um intervalo de tempo. Esta é a parte do Núcleo responsável por atribuir, alterar e executar estas animações. Este sub-módulo também é responsável por gerenciar a seqüência temporal de todas as animações de um projeto. Estas características serão detalhadas no capítulo 4.
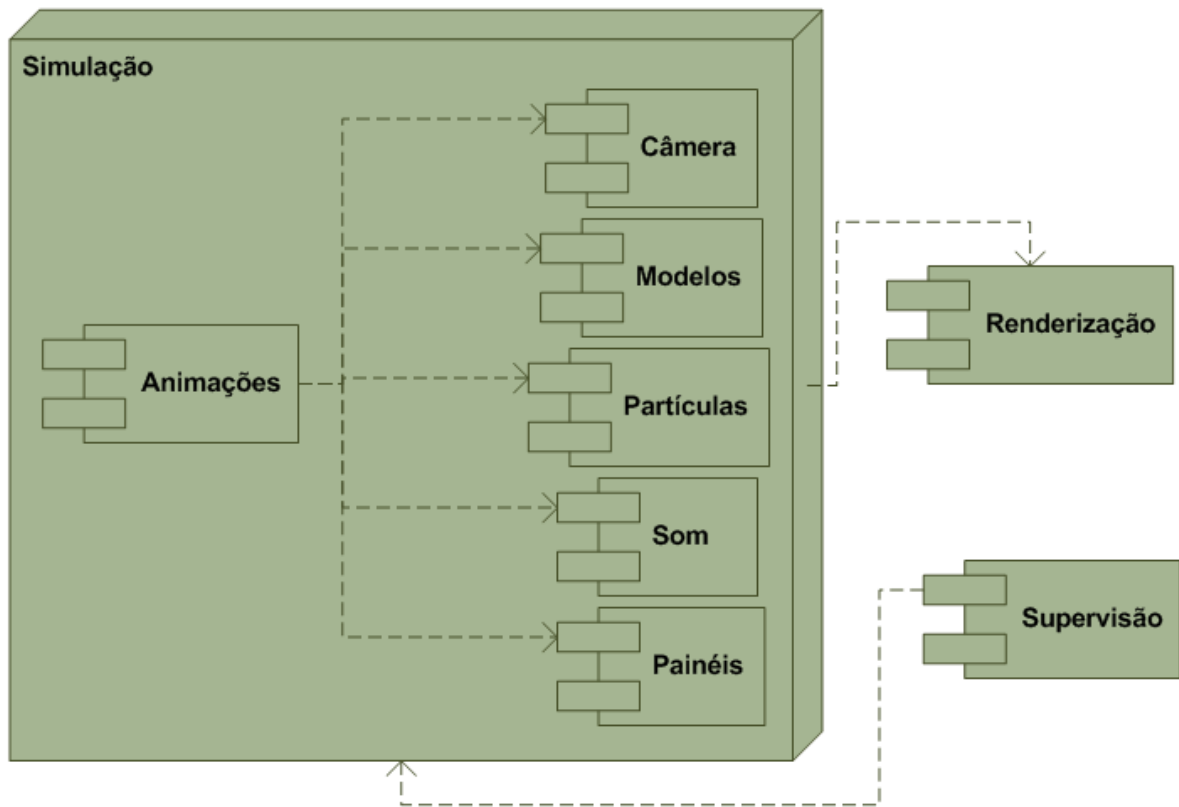
Figura 3.5 - Núcleo da arquitetura do Sistema de Autoria ITV

Renderização

O Núcleo também tem o papel de renderizar todas as mídias do Sistema de Autoria ITV, sendo a renderização de vídeo auxiliada principalmente pelas funções da biblioteca OGRE e as renderizações de som pela biblioteca FMOD. O papel da renderização é basicamente representar virtualmente os agentes no ambiente virtual, provocando a característica de imersão do usuário com a cena.

Supervisão

As simulações criadas pelo Sistema de Autoria ITV podem ser compostas de tarefas a serem realizadas pelo usuário. Ao executar estas tarefas o usuário será avaliado pelo sistema de supervisão autônoma, o qual faz parte do Núcleo da ITV.

3.3.2 GUI

A Interface Gráfica do Usuário (do inglês *Graphical User Interface*) do Sistema de Autoria ITV, assim como qualquer outra GUI, é o intermediador entre o usuário e o software e suas funções. Em particular neste software, além dos componentes convencionais de menus, botões e entradas de texto, a GUI possui um painel de exibição e interação gráfica 3D. A GUI do Sistema de Autoria ITV recebe ações do usuário através do mouse e teclado e exibe informações para ele seja por meio do ambiente virtual 3D (funções da biblioteca OGRE) ou de textos da GUI nativa do sistema operacional (componentes da biblioteca Qt).

Toda a interação no sentido do usuário para o sistema é gerenciada por funções da biblioteca Qt, inclusive interações no ambiente virtual, como cliques do mouse sobre modelos 3D. Isto se deve ao fato de que o verdadeiro “pintor” do ambiente virtual 3D é o Qt, pois a renderização da cena 3D feita pelo OGRE é transportada para a tela e “pintada” pela biblioteca Qt. Sendo assim não é possível interagir com a cena OGRE sem antes interagir com uma janela “pintada” pelo Qt.

3.3.3 Interpretador XML

Este módulo é responsável pela leitura e gravação de arquivos ITV, os quais estão em formato XML. Um mesmo arquivo ITV contém as informações de composição do grafo da cena 3D e a descrição de toda a simulação a ser realizada. Um arquivo ITV é dividido em três níveis principais:

- Configuração: define parâmetros como nome e descrição textual resumida da ITV, céu e cor de fundo do painel do ambiente virtual;
- Objetos: este nível contém todo o grafo de cena a ser montada pelo núcleo, com informações de posição, orientação, recurso em disco a ser carregado pelo módulo Carregador 3D, nome de exibição e quem é o seu objeto pai no grafo de cena;
- Passos: descreve todas as animações que compõe a simulação da ITV. O conceito de passos será explicado no capítulo 4.

Apesar de ser um documento inteligível a humanos, um arquivo de projeto ITV não necessita ser alterado manualmente pelo usuário, visto que o Sistema de Autoria ITV e o

módulo Interpretador XML se encarregam de traduzir todo documento para o ambiente virtual 3D e também de descrever todo o ambiente 3D e as animações em formato XML no momento de salvar uma ITV.

A figura 3.6 traz um arquivo de projeto ITV simplificado para fins didáticos.

```
1  - <xml>
2  -  <PROJETO>
3  -   <CONFIGURACAO>
4        <NOME>ITO_PARADA</NOME>
5       </CONFIGURACAO>
6  -   <OBJETOS>
7  -    <ger_rotor>
8         <MESH>ger_rotor.mesh</MESH>
9         <POSX>0</POSX>
10        <POSY>8.883</POSY>
11        <POSZ>0</POSZ>
12        <ROT>1 0 0 0</ROT>
13       </ger_rotor>
14      </OBJETOS>
15 -   <PASSOS>
16 -    <PASSO_0>
17        <NOME> Apresentação</NOME>
18        <DURACAO>20</DURACAO>
19        <TRANSICAO>3</TRANSICAO>
20        <FADE_IN>1</FADE_IN>
21        <OBJETOS_ROTACAO_INICIAL>Grupo@@ORIENTACAO_OBJETO&amp;&amp;1 0 0 0
22       </PASSO_0>
23      </PASSOS>
24     </PROJETO>
25   </xml>
```

Figura 3.6 - Documento ITV simplificado

3.3.4 Carregador 3D

O Sistema de Autoria ITV pode carregar modelos 3D, sons e efeitos visuais conhecidos como partículas que representam objetos do mundo real. Estas representações são armazenadas em disco nos mais variados formatos de arquivos. A parte do Sistema de Autoria ITV responsável pelo transporte destes arquivos para a memória volátil do computador é o Carregador 3D. O Carregador primeiro faz a catalogação de todos os arquivos contidos em uma pasta e suas sub-pastas sinalizando erros nos nomes dos arquivos, como nomes duplicados ou com acento. Carrega os arquivos que descrevem a texturização dos modelos 3D. Finalmente, a partir de uma ordem do Núcleo, o arquivo é carregado em memória pelo

Carregador e então exibido para o usuário. Abaixo estão listadas outras características importantes do Carregador 3D do Sistema de Autoria ITV:

- Controla o nível de detalhe dos modelos 3D a serem carregados em memória;
- Carrega modelos 3D no formato *mesh*;
- Carrega sons em diversos formatos, incluindo mp3, ogg e wav;
- Carrega partículas no formato descrito pela biblioteca OGRE;
- Responsável também pelo descarregamento da memória dos objetos.

Vale lembrar também o Sistema de Autoria 3D não pode criar nenhum dos arquivos carregados pelo Carregador 3D. Por exemplo, para adicionar um som ao projeto ITV, o usuário deve gravar um arquivo de som utilizando algum software de gravação sonora e posteriormente utilizar o arquivo no Sistema de Autoria ITV

4 IMPLEMENTAÇÃO DO SISTEMA DE AUTORIA DE ITV

O Sistema de Autoria ITV apresentado nesta dissertação representa um grande avanço em relação ao projeto UGV mostrado na seção 2.3 e principalmente em relação às instruções técnicas textuais utilizadas até então pela indústria. O Sistema de Autoria ITV proporciona um ambiente simples e totalmente livre de conhecimentos de programação para a construção de cenários, animações, simulações e treinamentos em realidade virtual 3D. Desta forma, um profissional da área conhecedor do processo real poderá confeccionar uma ITV que descreva aquele processo. Dentre algumas vantagens alcançadas com o Sistema de Autoria ITV pode-se destacar:

- A produção de uma ITV pode ser feita em grupo sem qualquer dificuldade na integração dos trabalhos;
- O software permite que sejam criadas inúmeras Instruções Técnicas Virtuais sem a necessidade de compilação;
- Pode-se modificar qualquer ITV pronta a fim de se adequar com o processo real;
- Pode-se aproveitar um trecho de uma ITV para compor outra;
- IT's textuais podem ser auxiliadas ou mesmo substituídas por ITV's.

Este capítulo descreve o Sistema de Autoria ITV e suas funções, destacando as ferramentas disponíveis para o usuário durante a construção de um ITV.

4.1 Interface

Toda a interface do Sistema de Autoria ITV foi projetada de forma a tornar o processo de construção de animações, simulações e treinamentos acessível a não-programadores. Sendo assim, a interface principal segue modelos de outros softwares populares, como Flash (TEAM, 2007), Power Point (MICROSOFT, 2010) ou mesmo outros sistemas de autoria como o NGrain, apresentado nesta dissertação. Esta característica é conhecida no estudo de Interfaces Homem-Máquina como consistência. Segundo (SPOLSKY, 2001), a consistência é a capacidade que uma interface possui de ser sintaticamente e semanticamente semelhante a outras interfaces já existentes. Por exemplo, um usuário está acostumado a fechar um

aplicativo com o atalho de teclado “Alt+F4”, ao atualizar o aplicativo para uma nova versão, descobre que o atalho para fechar agora é “Ctrl+S”. Certamente está mudança trouxe transtornos ao usuário porque o aplicativo não apresentou consistência em sua interface.

A interface do Sistema de Autoria ITV é composta de painéis, barra de menu, barra de ferramentas, listas hierárquicas, rótulos textuais, caixas de diálogo e um componente especial chamado de linha do tempo, o qual será descrito ainda neste capítulo.

A tela principal do Sistema de Autoria ITV é apresentada ao usuário logo após o carregamento de todo o núcleo e catalogação dos recursos. Porém este carregamento, que dura poucos segundos, é mascarado pela tela *splash* (painel com uma imagem de apresentação que fica sendo mostrado até que o programa seja totalmente carregado em memória) para dar um efeito de realimentação ao usuário, sinalizando que o programa está trabalhando.

Cada elemento da interface pode ter seu tamanho customizável, pode ser escondido temporariamente ou ainda pode ser colocado em uma janela separadamente da janela principal.

A figura 4.1 mostra a tela principal do Sistema de Autoria ITV com um cenário de exemplo já carregado.

Figura 4.1 - Tela principal do Sistema de Autotia ITV

Abaixo estão listadas as partes principais da interface do Sistema de Autoria ITV.

1. Ambiente 3D: é a principal área da interface, mostrando todo o cenário das animações renderizadas com objetos, partículas e informações textuais. O usuário também pode interagir com esta área através do mouse ou teclado, manipulando objetos e câmera.
2. Barra de menu: o usuário pode acessar algumas funções do Sistema de Autoria ITV a partir deste elemento. Os menus disponíveis são “Arquivo”, “Navegação”, “Ferramentas”, “Exibir” e “Configurações”.
3. Barras de Ferramentas: Contém os botões com as funções que permitem a produção de uma ITV. Estas funções vão desde mover um objeto até criação de uma nova ITV. As barras de ferramentas são “Arquivo”, ”Objetos”, “Manipulação”, “Animações”, “Projeto”, “Reprodução” e “Desfazer/Refazer”.
4. Árvore de objetos: Este elemento da interface mostra uma lista de todos os objetos da cena, sejam modelos 3D, grupos de objetos ou partículas. A lista é ordenada em ordem alfabética e hierarquizada de acordo com o grafo de cena do cenário 3D, ou seja, representa objetos pais agrupando objetos filhos na cena. O usuário também interage com este elemento, podendo alterar nomes de peças, alterar a hierarquia do grafo de cena ou selecionar uma ou mais peças do ambiente virtual.
5. Configuração da cena: Através deste elemento o usuário pode alterar parâmetros de objetos do cenário 3D, por exemplo: pode-se digitar a posição no cenário 3D das peças selecionadas ou mesmo alterar o nível de transparência de uma peça.
6. Configuração do passo: Pode-se configurar dois parâmetros fundamentais de um projeto ITV e seus passos, o tipo de transição e o tempo de duração do passo. O conceito de passos é explicado mais adiante neste capítulo.
7. Lista de passos: A lista de todos os passos do projeto ITV atual é mostrada neste componente da interface de modo que o usuário possa acessar qualquer passo diretamente apenas com um clique do mouse. Cada item da lista mostra o nome do passo e uma imagem do ambiente 3D naquele passo.
8. Linhas de tempo: A linhas de tempo são exclusivas de cada passo e de cada objeto da cena 3D, ou seja, cada objeto possui uma linha de tempo em cada passo. Através dela, pode-se alterar a animação dos objetos do cenário simplesmente mudando os marcadores da linha do tempo. O uso das linhas de tempo poderá ser mais bem entendido na seção 4.2.6.

4.2 Funcionamento de um projeto ITV

Os modelos 3D projetados em um programa separado do Sistema de Autoria ITV, como o Blender, podem ser carregados e configurados para formar a cena 3D inicial da ITV a ser construída. O sistema de Autoria ITV permite também carregar sistemas de partículas, alterar a hierarquia dos objetos da cena, agrupando-os para facilitar a manipulação, alterar parâmetros do objeto como orientação 3D, transparência, escala e diversas outras transformações, todas em tempo real. Para uma representação fiel da instrução técnica, o software conta com algumas funcionalidades que possibilitam a confecção de diferentes animações, simulações e treinamentos que, ao fim da geração da ITV, resultarão na descrição visual completa de um processo ou IT. Estas animações são dependentes do tempo e alteram o estado dos objetos da cena.

Cada uma destas animações pode estar em diferentes ou em um mesmo objeto ao mesmo tempo ou ainda, algumas animações especiais podem não estar vinculadas a objeto algum. A animação completa da ITV é composta de um ou mais passos. Estes passos são criados pelo usuário e possuem uma duração de tempo definida por este, de modo que ao término do tempo definido para um passo, o próximo passo é carregado e pode então ser executado com suas animações. Um passo é composto por um cenário inicial e por animações que irão modificar este cenário. O cenário inicial de cada passo é o estado de cada objeto, da câmera e do painel textual no início de um passo do projeto ITV. O estado de um objeto caracteriza-se por seu posicionamento, orientação, se possui ou não contorno, nível de transparência etc.

A transição entre dois passos adjacentes pode ser feita de três maneiras:

- Transição normal: o próximo passo é carregado automaticamente de modo que não se percebe a passagem de um passo a outro;
- Transição fade: ao fim de um passo ocorre escurecimento da cena e no início do próximo passo ocorre o clareamento da cena. Este efeito é útil para demonstrar descontinuidades na animação;
- Transição condicional: ao fim de um passo, o Sistema de Autoria ITV no modo de execução espera por uma interação do usuário (através do clique do mouse) com objetos do mundo virtual, de modo que, se o objeto correto for acionado, o próximo passo será carregado. Pode-se escolher um ou mais objetos da cena para serem os acionadores da transição.

O conceito de transição em um projeto ITV provê uma maior naturalidade às simulações, de modo que o usuário entende que o projeto ITV é executado como um todo. Além disso, o tipo de transição condicional tem grande importância na interação do usuário no decorrer da ITV. Por exemplo, se em um determinado processo uma bomba de óleo deve ser ligada manualmente pelo operador através do acionamento de uma chave. Este processo pode ser feito no Sistema de Autoria ITV criando-se uma transição do tipo condicional, que ocorrerá quando o usuário interagir com a ITV clicando no objeto 3D que representa a chave de ligação da bomba. Esta característica torna a experiência de aprendizado do processo mais prática e realista. A figura 4.2 descreve de forma simples a relação dos elementos envolvidos na construção de uma ITV. De acordo com a figura, cada passo Pn possui um cenário inicial e animações associadas. Essas animações ocorrem no intervalo de tempo TPn. Além disso, entre cada dois passos existe uma transição Tn para tornar a ITV concisa (BARATA, 2009).

ITV

Passo P1

Cenário inicial

Animações

Transição T1

Passo P2

Cenário inicial

Animações

Transição T2

...

Intervalo de tempo TP1

Intervalo de tempo TP2
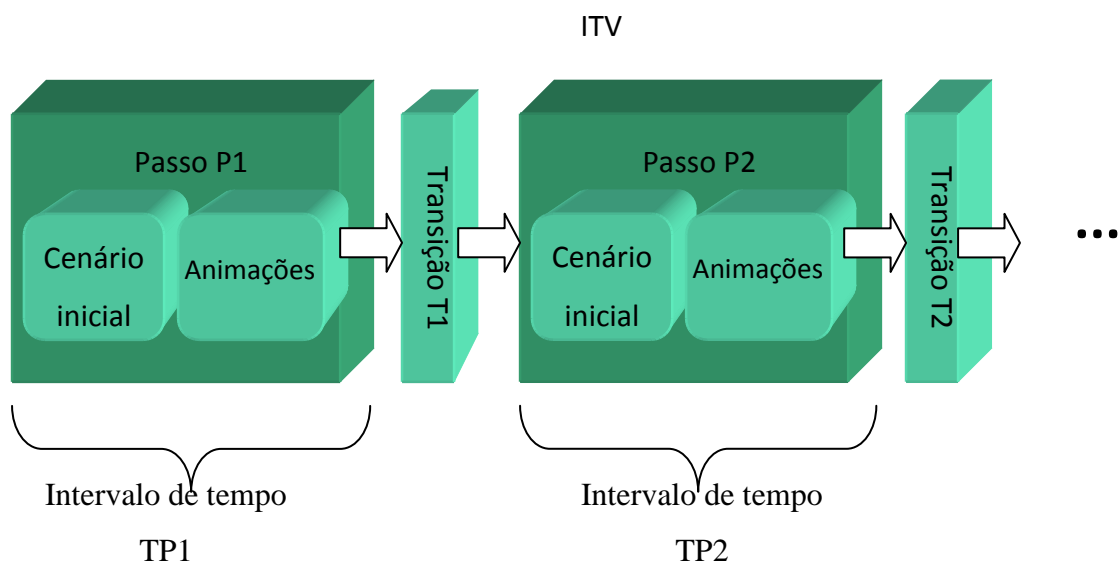
Figura 4.2 - Diagrama da estrutura de animações de uma ITV

4.2.1 Passos

Um projeto ITV é composto de passos seqüenciais. Pode-se pensar em um passo do projeto ITV como um *slide* de uma apresentação Power Point. Cada passo possui um tempo

de duração total, onde, durante este tempo, pode-se criar animações de todos os tipos. O conceito de passos é importante porque se aproxima do conceito de procedimentos da IT real, a qual possui também tarefas divididas em etapas bem definidas.

Além das animações, um passo também possui um cenário inicial, onde todos os objetos, painéis, sons e a câmera terão um estado único. Por exemplo, no passo sete de uma ITV o cenário inicial do objeto “gir_rotor.mesh” é na posição A, com rotação B e transparência C.

Em uma animação típica, o cenário inicial de um passo é o cenário final do passo anterior. O Sistema de Autoria ITV é capaz de pegar automaticamente o cenário final do passo anterior para utilizar no passo seguinte. Além disso, pode-se também, em um mesmo passo, inverter os cenários final e inicial. Por exemplo: no início de um passo um objeto está no ponto A (cenário inicial) e no final está no ponto B (cenário final). O Sistema de Autoria ITV possui a função de transformar o cenário inicial desse passo com o objeto no ponto B e o cenário final com o objeto no ponto A.

Porém, um passo não precisa necessariamente iniciar onde o anterior acabou. Por exemplo, a câmera está mostrando a chegada de um avatar no primeiro andar de um prédio, então o avatar pegar o elevador em direção ao sétimo andar, mas não se quer animar todo o trajeto do elevador entres esses andares. Então se faz um corte na animação de modo que no final de um passo a câmera esteja no primeiro andar e no início do passo seguinte ela já esteja no sétimo. Para situações como essa o produtor de uma ITV pode configurar a cena 3D como queira, posicionando a câmera ou alterando qualquer outro elemento do ambiente 3D e então memorizar o cenário inicial daquele passo manualmente apenas com um clique no botão.

A qualquer momento da produção de uma ITV um novo passo pode ser adicionado ao projeto em qualquer posição exceto na primeira posição, por exemplo, em uma ITV com sete passos pode-se inserir um novo passo entre os passos três e quatro ou após o sete, porém não se pode inserir um novo passo antes do primeiro. Esta exceção não representa uma limitação do Sistema de Autoria ITV, mas sim um conceito a ser seguido: o primeiro passo representa o estado inicial de todos os elementos do projeto, seja modelos 3D, câmera, painel textual ou som. Portanto se o produtor da ITV quiser criar um novo primeiro passo, ele deve na verdade alterar o estado dos elementos do primeiro passo.

O Sistema de Autoria ITV pode ainda copiar um passo já existente para outro passo qualquer. Desta forma podem-se aproveitar animações e cenários já criados para compor a ITV, aumentando a produtividade à medida que se reaproveita passos já feitos.

A figura 4.3 mostra os botões de adição de passo, remoção de passo, memorização do cenário inicial, cópia do cenário final do passo anterior, cópia de passos, inversão de cenários em ordem da esquerda para a direita.

Figura 4.3 - Botões de manipulação de passos

A lista de todos os passos de uma ITV é mostrada no painel Lista de passos, apresentado na figura 4.4. Sendo que cada passo possui um nome e uma imagem representando-o. Além disso, a seleção de qualquer um desses ícones na lista de passos fará com que o passo referente seja carregado e mostrado no Sistema de Autoria ITV, podendo então ser executado ou alterado.
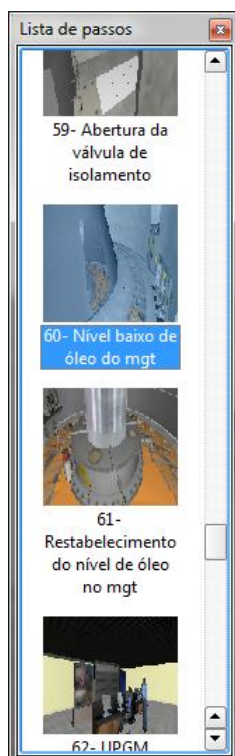
Figura 4.4 - Lista de passos do Sistema de Autoria ITV

A qualquer momento da produção de uma ITV, o usuário do sistema pode alterar o tempo de duração do passo corrente, sem perder o trabalho já feito nele. O usuário pode escolher entre adicionar mais tempo ao passo, mantendo as animações intactas ou pode “esticar” ou “encolher” o tempo, “esticando” ou “encolhendo” na mesma proporção o tempo das animações do passo que teve o tempo alterado.

4.2.2 Carregamento

O objetivo do Sistema de Autoria ITV é a criação animações envolvendo modelos 3D para representar simulações e treinamentos reais. Porém o software não é um modelador 3D. Desta forma é necessário que se utilize de outros programas de autoria para criação dos modelos que serão utilizados nas ITV’s. O Blender, um software de modelagem 3D gratuito e de código aberto, um dos modeladores mais utilizados no mundo (HESS, 2007), é um exemplo de software utilizado pra este fim.

O Sistema de Autoria ITV pode carregar objetos no formato *mesh* com animações embutidas no formato *skeleton,* padronizado pela plataforma OGRE. Ambos estes formatos são facilmente exportados a partir de formatos mais conhecidos como o 3ds do 3D Studio Max. Para isso, basta usar um dos vários exportadores disponíveis na comunidade OGRE. Pode-se ainda criar um objeto ou uma animação embutida diretamente para os formatos OGRE através do Blender. O tipo de animação *skeleton* é feita em tempo de modelagem e também pode ser carregada pelo Sistema de Autoria ITV; É usada quando a animação que se pretende fazer envolve alteração na malha de modelagem. Por exemplo, uma animação de um avatar andando altera a forma da malha geométrica modelada, pois simula ossos e suas articulações, este tipo de animação deve ser feita em tempo de modelagem e depois carregada no Sistema de Autoria ITV. Mais sobre animação do tipo *skeleton* na seção 4.2.4.

O carregamento dos objetos *mesh* no Sistema de Autoria ITV pode ser feito de duas maneiras:

O primeiro modo de carregamento é utilizado quando se quer montar uma cena completa, ou seja, além de carregar um ou mais objetos no ambiente, o Sistema de Autoria ITV também configura os parâmetros de posição e orientação deles. Para isso deve-se usar uma ITV já pronta (O Sistema de Autoria ITV vai utilizar o cenário inicial do primeiro passo) ou um script XML como mostrado na figura 4.5.

```
<xml>
  <grd_parafuso_base_aste_gda01>
   <NOME>grd_parafuso_base_aste_gda01</NOME>
   <TIPO>MESH</TIPO>
   <MESH>grd_parafuso_base_aste_gda01.mesh</MESH>
   <PARTICULA></PARTICULA>
   <PAI></PAI>
   <POSX>0.676288</POSX>
   <POSY>10.1783</POSY>
   <POSZ>8.32735</POSZ>
   <ROT>1 0 0 0</ROT>
  </grd_parafuso_base_aste_gda01>
</xml>
```

Figura 4.5 - Estrutura XML que pode ser usada para carregamento de objetos

O segundo modo é um carregamento de objetos no ponto 3D de origem do ambiente virtual. Neste modo as peças carregadas deverão ser manipuladas manualmente a fim de montar a cena.

Em ambos os modos, o usuário pode adicionar um ou mais objetos de uma vez através de uma listagem de todos os arquivos *mesh* contidos nas pastas configuradas no Sistema de Autoria ITV. No momento do carregamento escolhe-se ainda o nível de detalhe (do inglês *Level of Detail - LoD*) que a peça carregada vai assumir sendo que quanto maior o valor escolhido mais simplificado será o modelo. O *LoD* é uma técnica para melhorar a performance de aplicações 3D aplicando-se técnicas de redução da geometria dos modelos 3D em tempo de renderização. Além disso, após a escolha dos objetos a serem carregados, o sistema informa a progressão do carregamento, de forma que o usuário receba um *feedback*, do funcionamento do programa. É obvio que o Sistema de Autoria ITV também permite a exclusão de objetos da cena, o que fará com que todas as animações dos objetos excluídos também sejam removidas do projeto. A figura 4.6 mostra o painel que lista os objetos para carregamento.
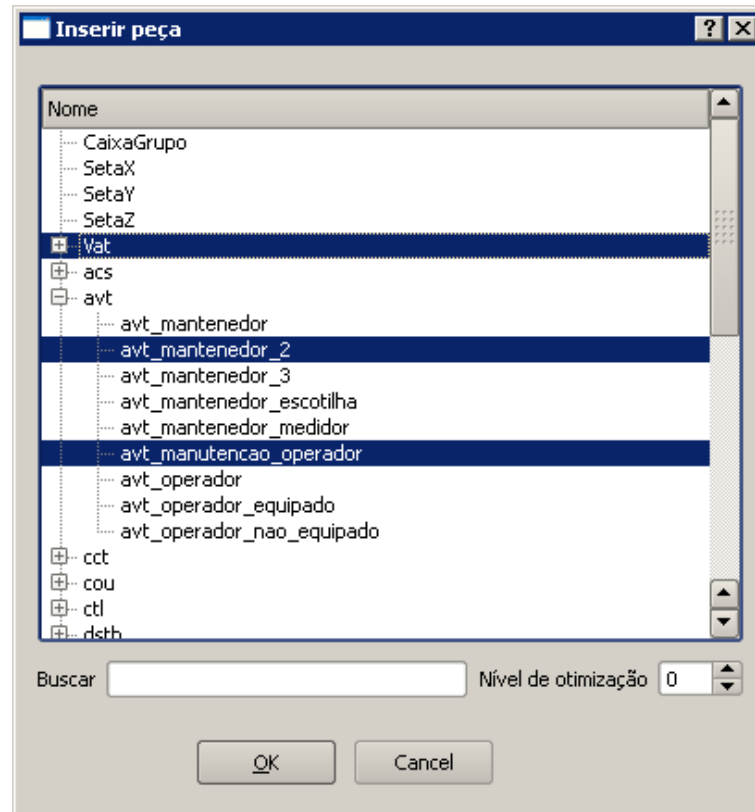
Figura 4.6 - Listagem dos objetos disponíveis para carregamento na cena

O Sistema de Autoria ITV também é capaz de adicionar partículas à cena 3D. Um sistema de partículas é uma técnica não convencional de renderização 3D (não utiliza malha geométrica) para representar efeitos difusos, como chuva, fogo e espuma. A figura 4.7 apresenta um exemplo de sistema de partículas de água, carregado no Sistema de Autoria ITV.
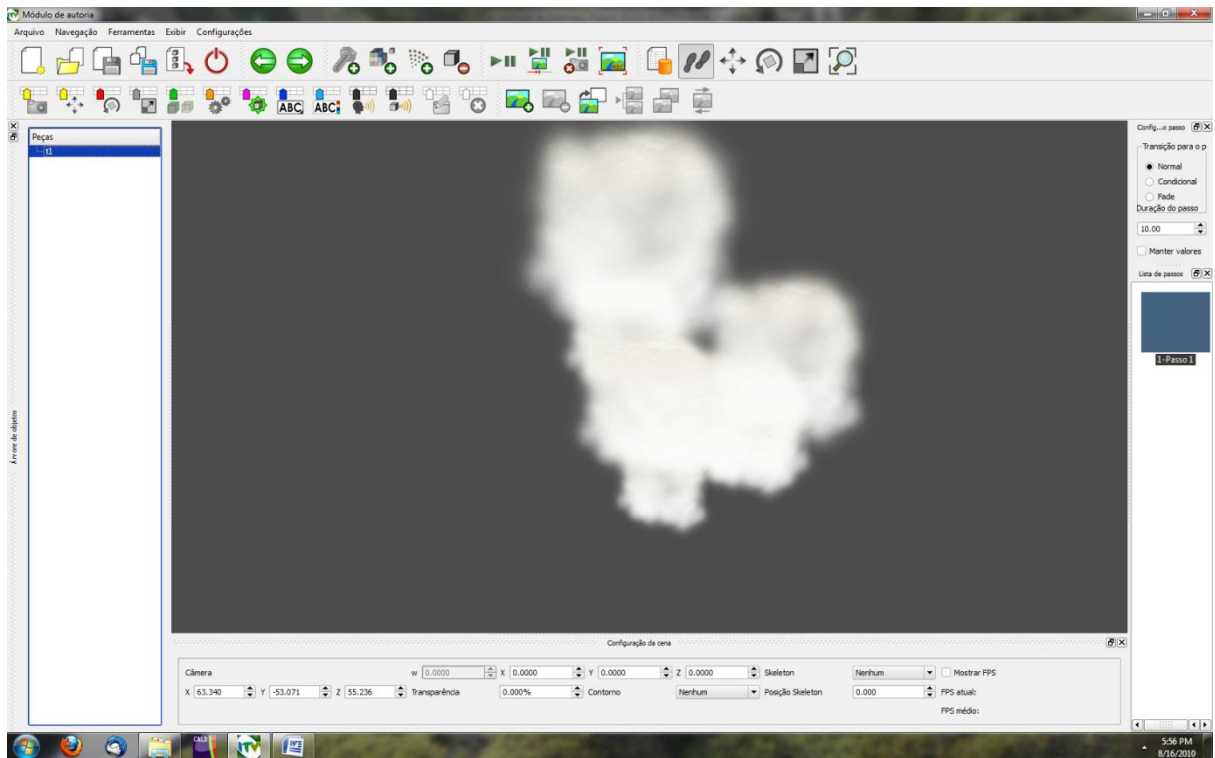
Figura 4.7 - Exemplo de sistema de partículas no Sistema de Autoria ITV

O Sistema de Autoria ITV possui um componente que lista todos os objetos contidos na cena, a árvore de objetos. Este componente, mostrado na figura 4.8, serve como uma forma prática de selecionar um ou mais objetos ao mesmo tempo. A seleção na árvore de objetos reflete na seleção dos objetos no ambiente virtual, assim como a seleção dos objetos no ambiente virtual reflete na árvore de objetos.
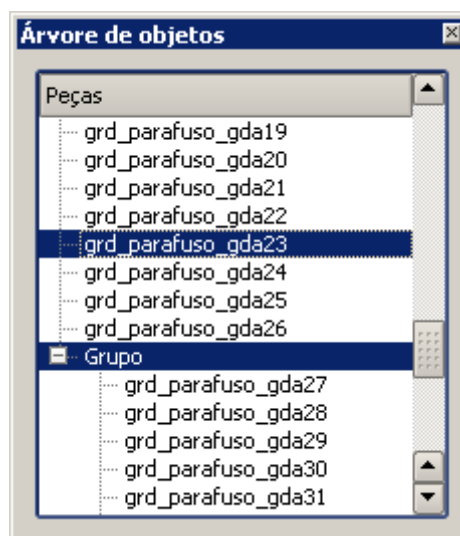
Figura 4.8 - Árvore de objetos do Sistema de Autoria ITV

Existe ainda um tipo especial de objeto que é invisível: o objeto agrupador não é mostrado na cena 3D, mas pode ser visualizado na árvore de objetos, servindo apenas para compor hierarquias no grafo. Por exemplo, a figura 4.8 mostra o objeto “Grupo” que está um nível de hierarquia acima do objeto “grd_parafuso_gda27”, porém está no mesmo nível de “grd_parafuso_gda23”. Diz-se então que “Grupo” é pai de “grd_parafuso_gda27” e irmão de “grd_parafuso_gda23”. O Sistema de Autoria ITV permite que a hierarquia seja feita em vários níveis, por exemplo, “Grupo” é pai de “grd_parafuso_gda27” e filho de “Grupo_P”. Apesar de ser invisível no ambiente virtual, agrupadores podem receber qualquer animação do Sistema de Autoria ITV. Dessa forma, podem-se simular movimentos que seriam impossíveis de serem feitos sem usar hierarquia, a exemplo do movimento de translação da terra em torno do sol.

4.2.3 Transição

Uma transição no Sistema de Autoria ITV nada mais é do que a forma com que um passo n é finalizado para o início do passo n+1. A transição é importante para que o projeto ITV como um todo tenha uma seqüência coerente.

Transição normal

Neste tipo de transição o passo é transicionado automaticamente sem qualquer efeito especial, ou seja, simplesmente o passo n+1 é carregado ao final do passo n. A transição normal é indicada para situações em que a animação do próximo passo é uma perfeita seqüência da animação do passo atual, por exemplo, quando em uma animação do passo n a câmera finaliza posicionada no ponto A e no passo n+1 a animação da câmera inicia no ponto A.

Transição Fade

A transição Fade adiciona um efeito especial na mudança entre os passos. Este efeito é caracterizado pelo escurecimento total e gradativo do ambiente virtual no fim do passo n e clareamento gradativo e total no inicio do passo n+1. Este feito é demonstrado na figura 4.9.
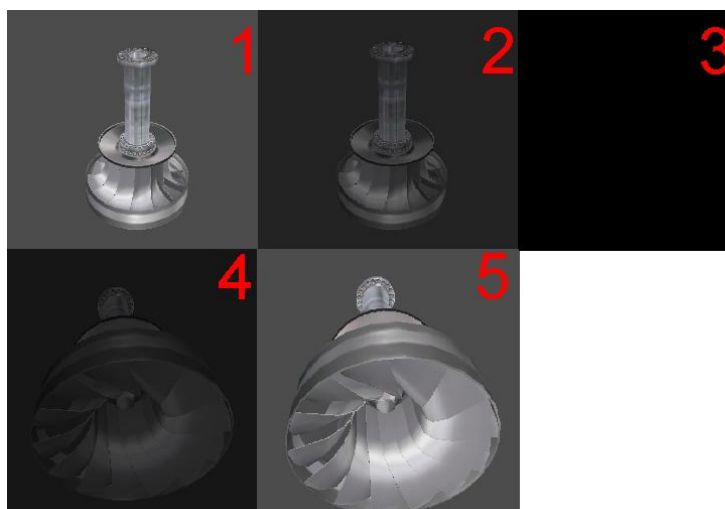
Figura 4.9 – exemplo de seqüência de transição Fade

A transição fade, ao contrário da transição normal, é indicada para animações descontínuas, de modo que a transição entre elas não seja brusca, evitando desconfortos para quem usa a ITV pronta.

Transição condicional

Em ITV’s de treinamento o conceito mais importante da realidade virtual é a interação, de modo que o usuário interaja com a cena 3D a fim de realizar uma tarefa. No

Sistema de Autoria ITV existe a transição condicional, como uma forma muito simples e prática para agregar interações à ITV's feitas com o propósito de treinamento.

Para exemplificar, imagina-se um treinamento em realidade virtual onde o treinando deve saber em que momento puxar as alavancas e os botões corretos de um painel de controle de um processo de beneficiamento de madeira. Seria muito útil neste treinamento que o usuário não só olhasse os botões e alavancas sendo acionados em uma animação, mas que ele mesmo pudesse acionar os comandos interagindo no ambiente virtual. No Sistema de Autoria ITV o produtor pode usar a transição condicional para criar treinamentos deste tipo.

A transição condicional faz com que a mudança de um passo n para o passo seguinte n+1 ocorra somente quando o usuário clique com o mouse em um dos modelos 3D escolhidos como objetos de transição. Esta escolha é feita durante a produção da ITV.

4.2.4 Manipulação

O Sistema de Autoria ITV possibilita que os objetos do cenário 3D sejam manipulados de diversas formas. A manipulação dos objetos é o ato do usuário ou do produtor da ITV de alterar parâmetros a fim de explorar ou montar a Cena 3D ou ainda criar animações com a ajuda das manipulações.

Podem-se alterar os seguintes objetos: modelos 3D, grupos, câmera e sistema de partículas. No objeto câmera pode-se alterar a sua posição e a sua direção. Em todos os outros se pode alterar a posição, a orientação, a escala, a transparência, a marcação de cor e a animação embutida.

Ponto de pivô

O conceito de ponto pivô foi herdado dos programas de modelagem 3D e é usado no Sistema de Autoria ITV para ações de manipulação e também animações. O ponto de pivô é o ponto no espaço em torno do qual todas as transformações como rotações e escalas são centrados. Por exemplo, para rotacionar o ponteiro de um relógio, o seu ponto de pivô deve estar na extremidade inferior do ponteiro. Ou ainda, para criar uma animação de rotação num pneu de um carro, o ponto de pivô deve estar no centro do pneu.

No Sistema de Autoria ITV o ponto de pivô não é configurável, porém ele herda o mesmo ponto de pivô do modelo 3D *mesh* durante a modelagem.

Ferramenta pivô

A ferramenta pivô do Sistema de Autoria ITV permite que manipulações de posição, orientação ou escala sejam feitas com o mouse. Ela é mostrada quando um ou mais objetos são selecionados. Sua orientação e escala são sempre as mesmas independentemente dos objetos selecionados ou da visão da câmera, porém a posição da ferramenta pivô depende dos objetos selecionados, sendo que estará no ponto pivô do objeto quando somente um estiver selecionado, ou, quando houver mais de um selecionado, estará no centro dos objetos. A figura 4.10 mostra a ferramenta pivô quando somente um objeto está selecionado na cena.
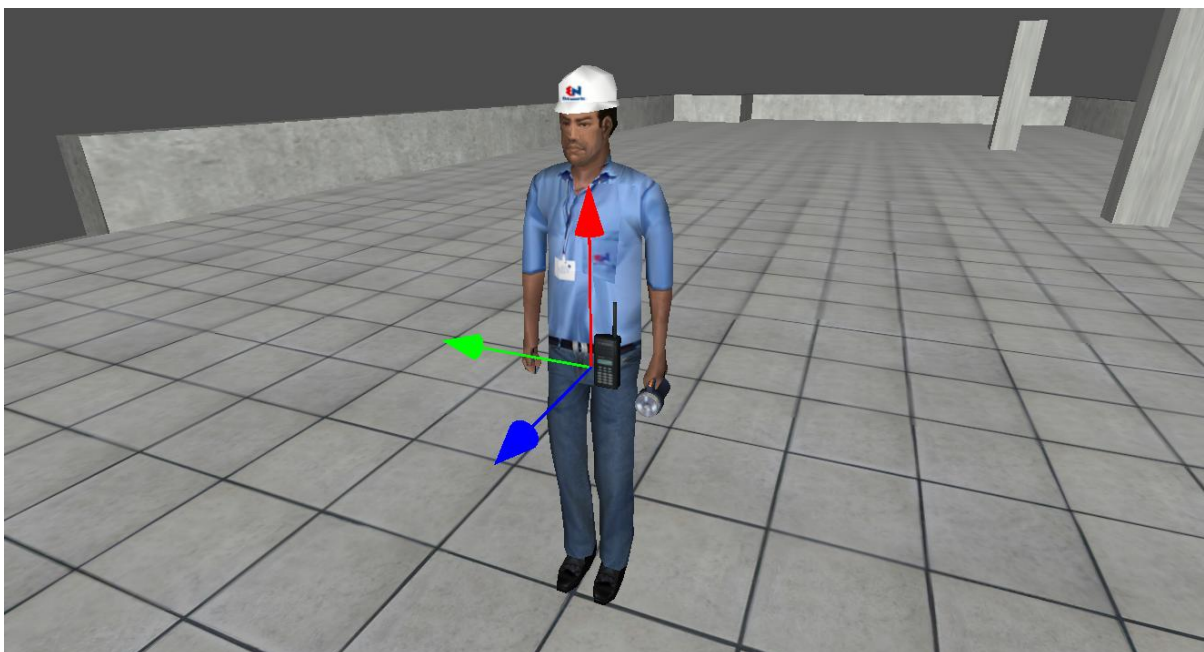
Figura 4.10 - Ferramenta pivô do Sistema de Autoria ITV

A ferramenta tem a forma de três setas coloridas representando os eixos das três dimensões: seta verde representa o eixo X, seta vermelha representa o eixo Y, seta azul representa o eixo Z. Além disso, a ferramenta sempre é visível quando há algum objeto selecionado, por mais que este objeto não esteja visível no momento.

Posição

É o ponto na cena 3D que o objeto está localizado, ou melhor, é o onde, na cena 3D, o ponto pivô do objeto está localizado. Este ponto é definido em três dimensões, sendo designados como sendo ponto X, Y e Z. O Sistema de Autoria ITV pode alterar a posição de um ou mais objetos na cena utilizando a ferramenta pivô ou digitando o valor dos pontos X, Y ou Z no painel de configuração da cena.

Orientação

O conjunto de rotações de um objeto em torno dos próprios eixos do ponto pivô forma a orientação desse objeto. No Sistema de Autoria ITV também é possível alterar a orientação a partir do mouse usando a ferramenta pivô, ou ainda se pode alterar os valores de orientação, expressos em Quaternion (conjunto de números comumente usados em matemática 3D, (WEISSTEIN, 2010)), no painel de configuração da cena. A figura 4.11 é um exemplo de uma alteração na orientação do objeto da figura 4.10. Neste exemplo ocorreu uma rotação do objeto em torno do eixo Y.
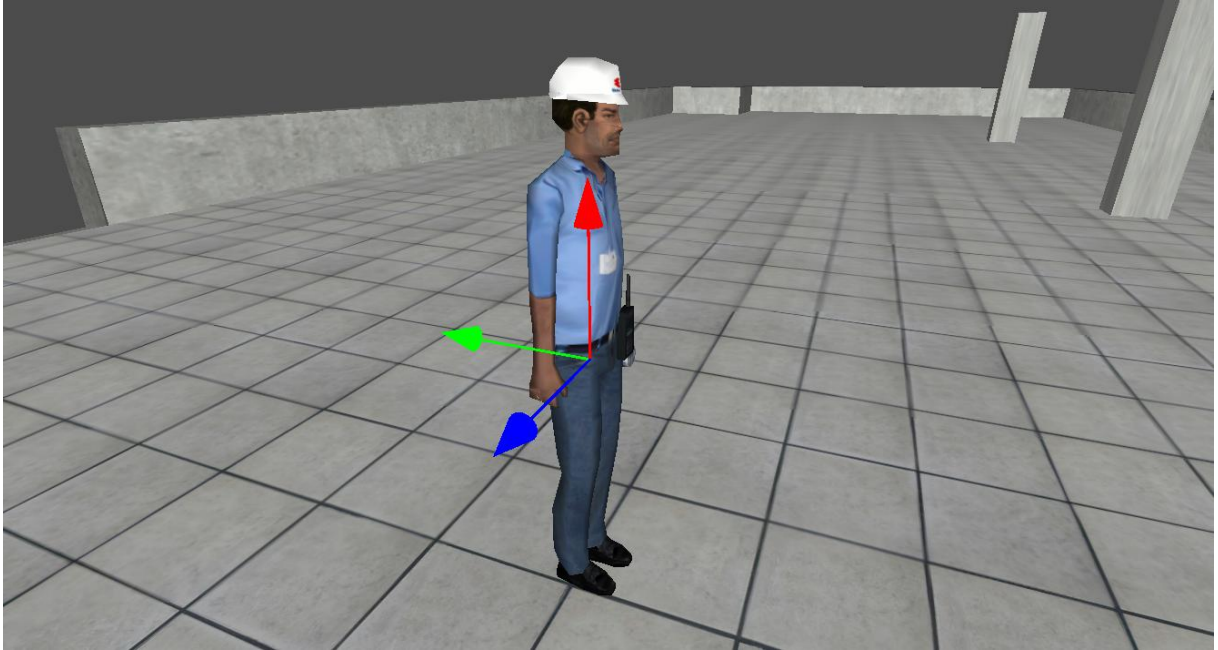
Figura 4.11 - Exemplo de manipulação da orientação no Sistema de Autoria ITV

Escala

A manipulação de escala dos objetos significa manipular as dimensões deles. Mais uma vez, esta manipulação pode ser feita com a ferramenta pivô ou através do painel de configuração da cena alterando-se os valores multiplicadores das escalas em cada eixo. A figura 4.12 mostra o objeto da figura 4.10 com suas dimensões alteradas usando-se a manipulação de escala.
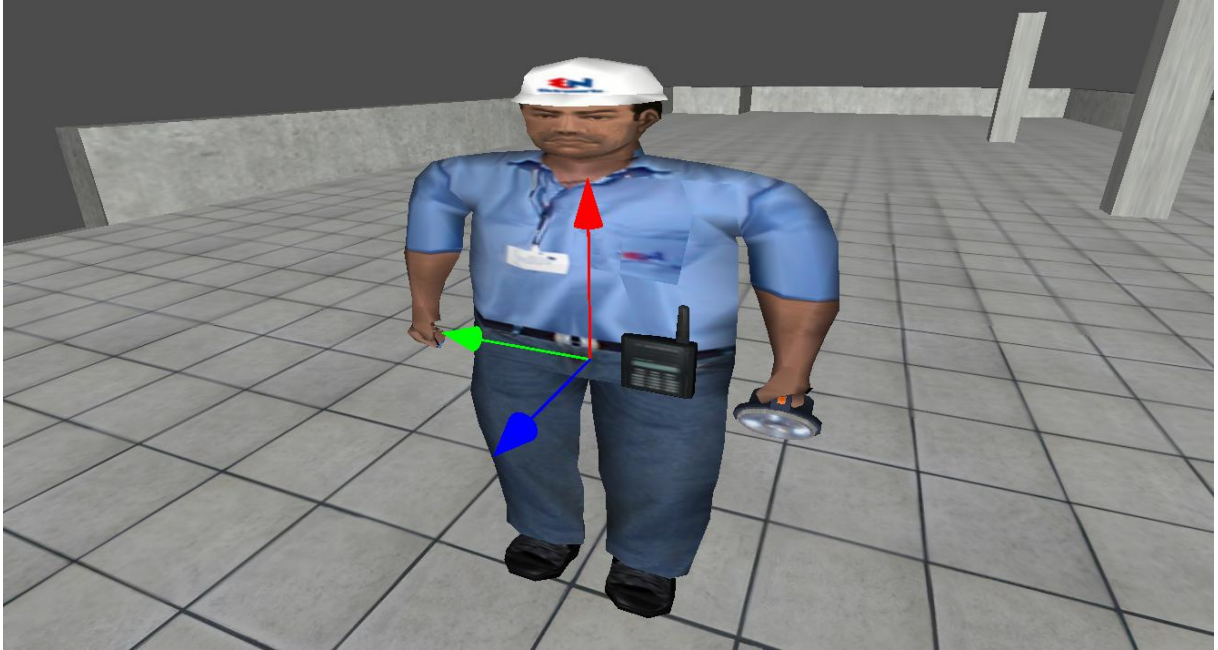
Figura 4.12 - Exemplo de utilização da manipulação de escala

Câmera

A câmera do Sistema de Autoria ITV são os olhos do usuário sobre o ambiente virtual, ou seja, a cena renderizada e mostrada no painel do ambiente virtual é aquela cena para a qual a câmera está apontada. A câmera, assim como outros objetos da cena, pode ser manipulada para que o usuário veja as partes que se deseja no ambiente virtual.

O Sistema de Autoria ITV pode manipular a câmera de três maneiras diferentes podendo ser usadas simultaneamente: usando a ferramenta “Andar” que permite ao usuário mover a câmera para qualquer ponto e apontá-la para qualquer direção somente usando o mouse; usando o teclado para manipular a câmera completamente; digitando no painel de configuração da cena a posição da câmera e o seu ponto de foco na cena 3D.

Transparência

Um objeto é dito transparente quando é possível ver através das faces da sua geometria. O Sistema de Autoria ITV pode manipular totalmente a transparência de qualquer objeto da cena 3D, inclusive sistema de partículas, tornando-o totalmente transparente (invisível) ou nada transparente. Esta manipulação é muito útil para demonstrar partes ocultas

dos modelos representados, por exemplo, tornar transparente a representação dos músculos de um avatar para visualizar o seu esqueleto. Pode-se também escolher qualquer valor intermediário de transparência.

A manipulação de transparência no Sistema de Autoria ITV só pode ser feita pelo painel de configuração da cena. A figura 4.13 mostra o objeto da figura 4.10 com cinqüenta por cento de transparência.
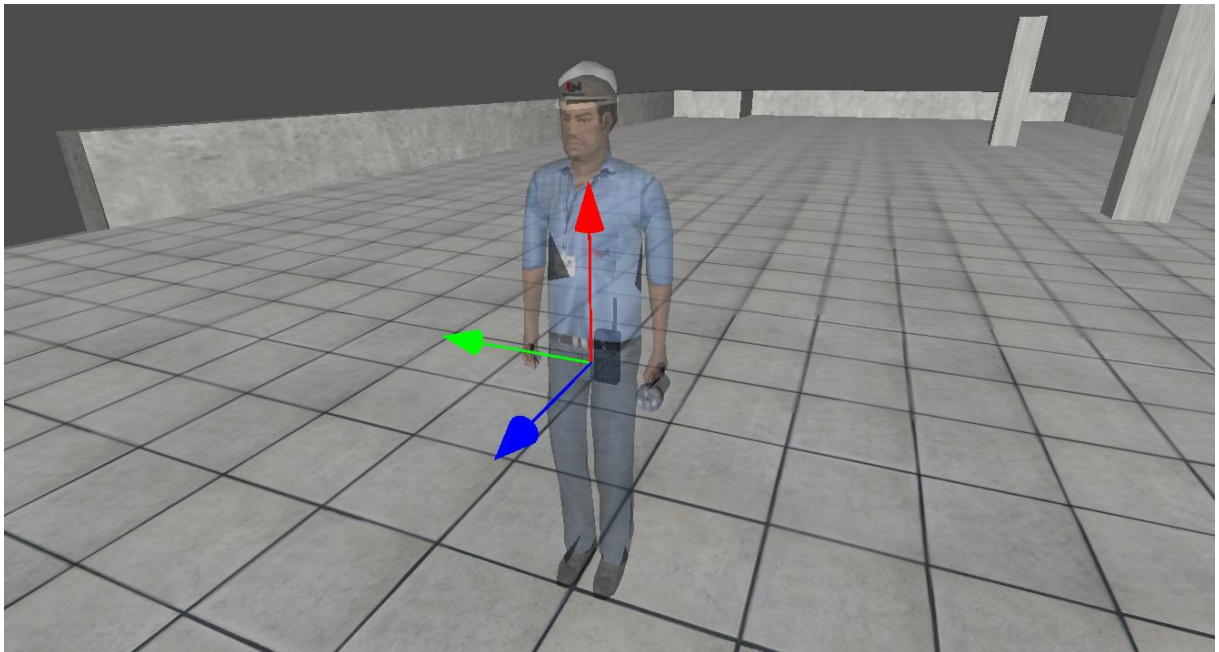
Figura 4.13 - Exemplo de aplicação de transparência sobre um objeto da cena 3D

Marcação de cor

O efeito de marcação de objetos no Sistema de Autoria ITV é uma forma de destacar certo objeto dos demais na cena 3D. O usuário pode escolher a cor de marcação através do painel de configuração da cena e então o objeto ficará “piscando” na cor escolhida. A figura 4.14 apresenta o objeto da figura 4.10 marcado com a cor verde.
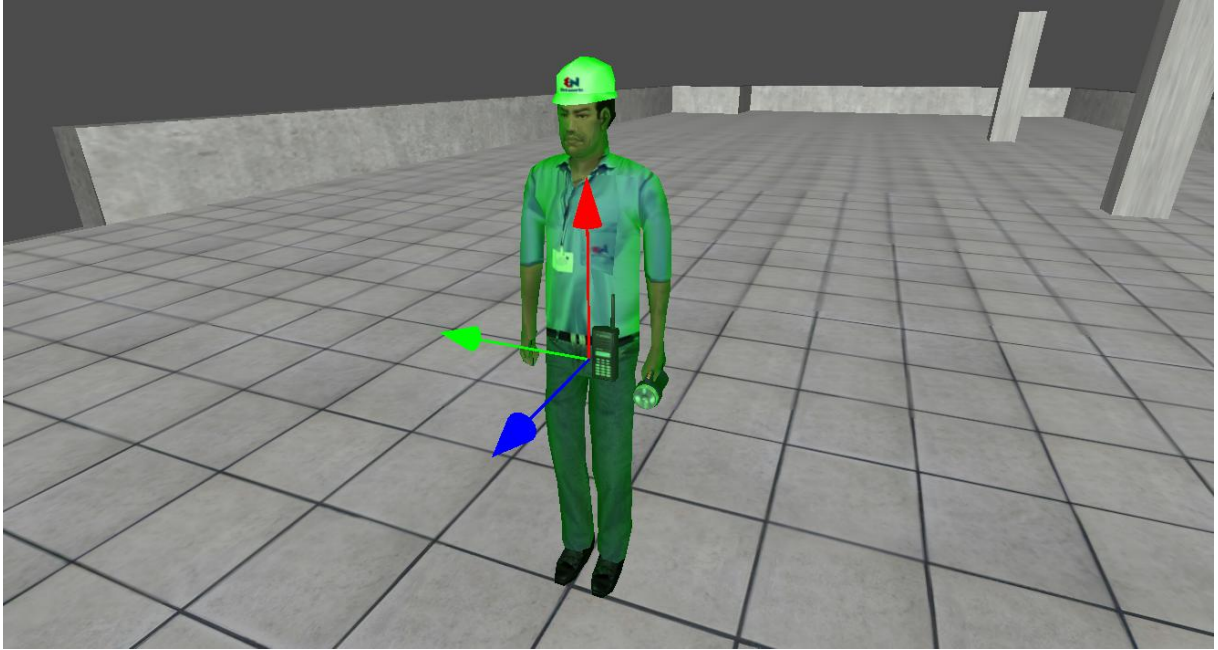
Figura 4.14 - Exemplo de efeito de marcação sobre um objeto

Animação embutida

O Sistema de Autoria ITV pode carregar animações embutidas para os modelos 3D. Estas animações devem ser exportadas em tempo de modelagem no formato *skeleton*. Este tipo de animação é recomendado quando a geometria do modelo 3D irá sofrer modificação. O software permite que se escolha uma animação embutida dentre um conjunto de animações disponíveis, além de permitir escolher o momento da animação. A figura 4.15 apresenta o modelo da figura 4.10 com uma animação embutida.
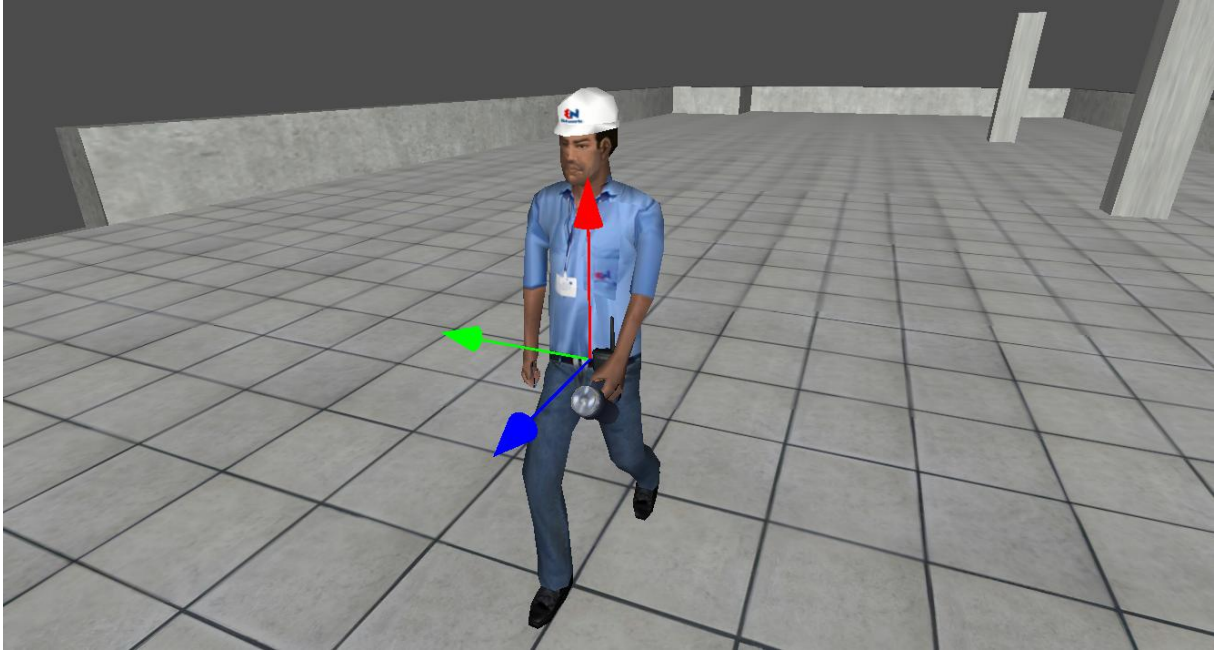
Figura 4.15 - Exemplo de objeto com animação embutida

4.2.5 Animações

Uma das principais características do Sistema de Autoria ITV é criar animações nos objetos da cena 3D carregada, de forma simples e sem conhecimentos de programação. Para cumprir este fim, o sistema dispõe de um grande conjunto de tipos de animação, sendo que algumas animações são exclusivas de alguns tipos de objetos.

As animações podem durar no máximo o tempo de duração do passo em que elas estão, sendo que em outro passo pode existir uma animação do mesmo tipo e para o mesmo objeto. Cada animação pode ser configurada a partir de uma caixa de diálogo com parâmetros de cada animação.

Texto no ambiente virtual

A inserção de texto na animação é útil para explicações do procedimento ocorrido no momento, porém é obvio que este tipo de animação pode ser utilizado de inúmeras formas. A figura 4.16 apresenta uma caixa de diálogo para configuração da animação de texto no ambiente virtual.
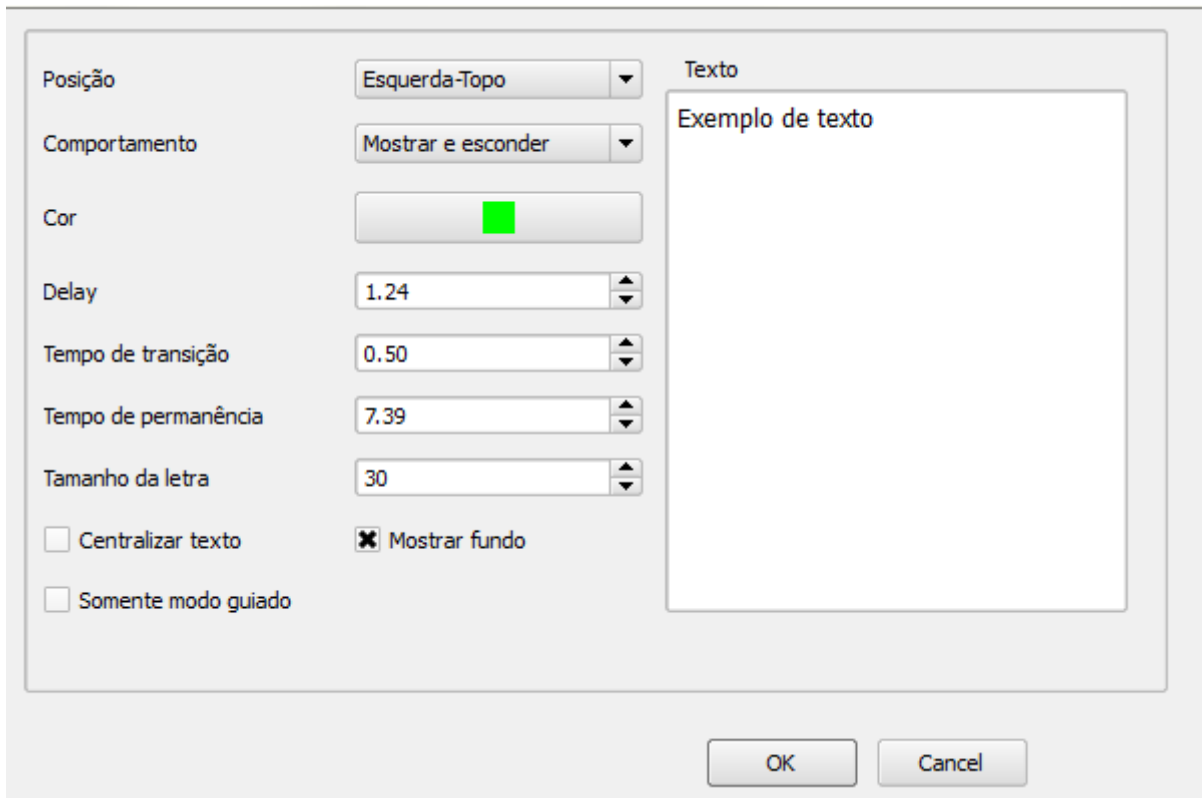
Figura 4.16 - Caixa de configuração da animação de texto do ambiente virtual

A animação possui um efeito de transparência sempre que o texto vai ser mostrado ou escondido, de modo que não seja uma transição brusca, além disso, pode-se escolher o tamanho da letra, a cor de fundo do texto, posição do texto no ambiente virtual e outras configurações. A figura 4.17 mostra um exemplo deste tipo de animação.

Figura 4.17 - Exemplo de texto no ambiente virtual

Texto na interface

Outra animação que pode ser utilizada na produção de ITV's é a apresentação de textos fora do ambiente virtual, ou seja, o texto é mostrado em um painel separadamente da cena 3D. O painel também é mostrado com um efeito de transparência, porém não se pode escolher a posição do texto ou a cor dele. A figura 4.18 mostra um exemplo de texto fora do ambiente virtual
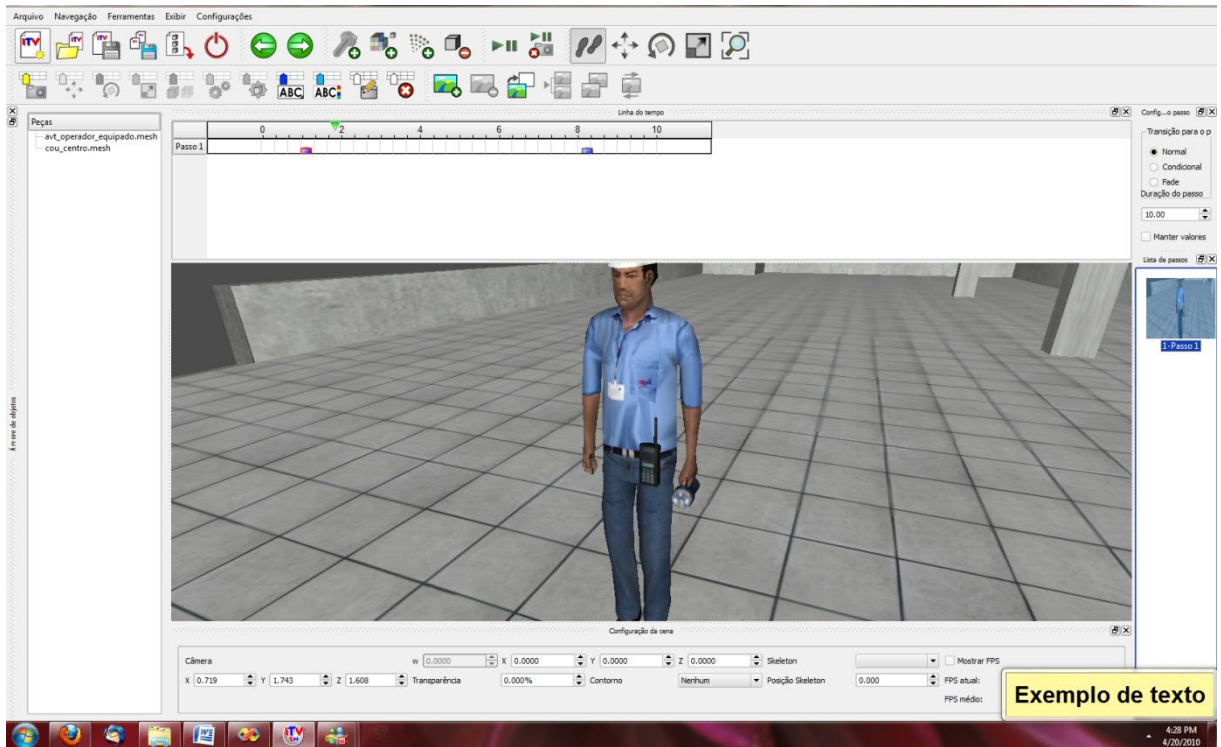
Figura 4.18 - Exemplo de animação com texto apresentado fora do ambiente virtual

Translação de objeto

Quando se quer animar o movimento de um objeto de um ponto A para um ponto B, este tipo de animação pode ser utilizado. Na verdade, o Sistema de Autoria ITV permite a criação de inúmeros pontos na trajetória da animação, onde o software irá interpolar o caminho descrito pelos pontos definidos. Este tipo de animação pode ser feito de maneira rápida com o mouse, definindo-se os pontos da animação no próprio ambiente virtual. A figura 4.19 demonstra uma animação de translação, onde os objetos semitransparentes representam a trajetória seguida durante a animação entre somente dois pontos.

Figura 4.19 - Demonstração da animação de objetos do Sistema de Autoria ITV

Escala de objetos

Outra possibilidade de animação do Sistema de Autoria ITV é a escala de objetos. Com esta animação pode-se alterar o tamanho dos objetos de forma gradativa. Escolhem-se diferentes escalas de um objeto no tempo e o sistema anima a transformação em escala nestes tempos definidos. Esta animação também pode ser criada com o mouse. A figura 4.20 representa uma animação de escala.

Figura 4.20 - Animação de escala no Sistema de Autoria ITV

Rotação de objetos

A animação de rotação de um objeto consiste na animação da transformação de orientação de um objeto qualquer da cena. Pode-se definir um eixo qualquer de rotação do objeto ou adotar automaticamente o eixo pré-definido em tempo de modelagem, a velocidade de rotação, a aceleração da rotação e é claro, o tempo de rotação dentro do passo. Além disso, pode-se rotacionar um objeto agrupador, criando assim animações do tipo translação em círculo. A figura 4.21 é uma representação da rotação de um objeto usando a animação do Sistema de Autoria ITV.

Figura 4.21 - Animação de rotação

Animação de câmera

A animação da câmera pelo ambiente virtual também pode ser criada no Sistema de Autoria ITV. Para isto, o criador do projeto ITV navega com a câmera pela cena e escolhe duas ou mais visões do ambiente, ou seja, escolhe posições e direções da câmera simplesmente visualizando o ambiente virtual. Quando a animação for reproduzida, o Sistema de Autoria ITV se encarrega de criar a animação entre os pontos escolhidos pelo projetista. A figura 4.22 mostra vários quadros com visões da câmera no ambiente virtual, sendo que os quadros das extremidades são os pontos escolhidos pelo usuário. Vale lembrar que o usuário pode escolher infinitos pontos de visão da câmera dentro de um mesmo passo e que os quadros da interpolação do ITV são criados em uma taxa de sessenta por segundo, de modo que usuário perceba uma mudança suave de visão. Esta taxa de atualização vale também para todas as outras interpolações ocorridas nas animações do Sistema de Autoria ITV.
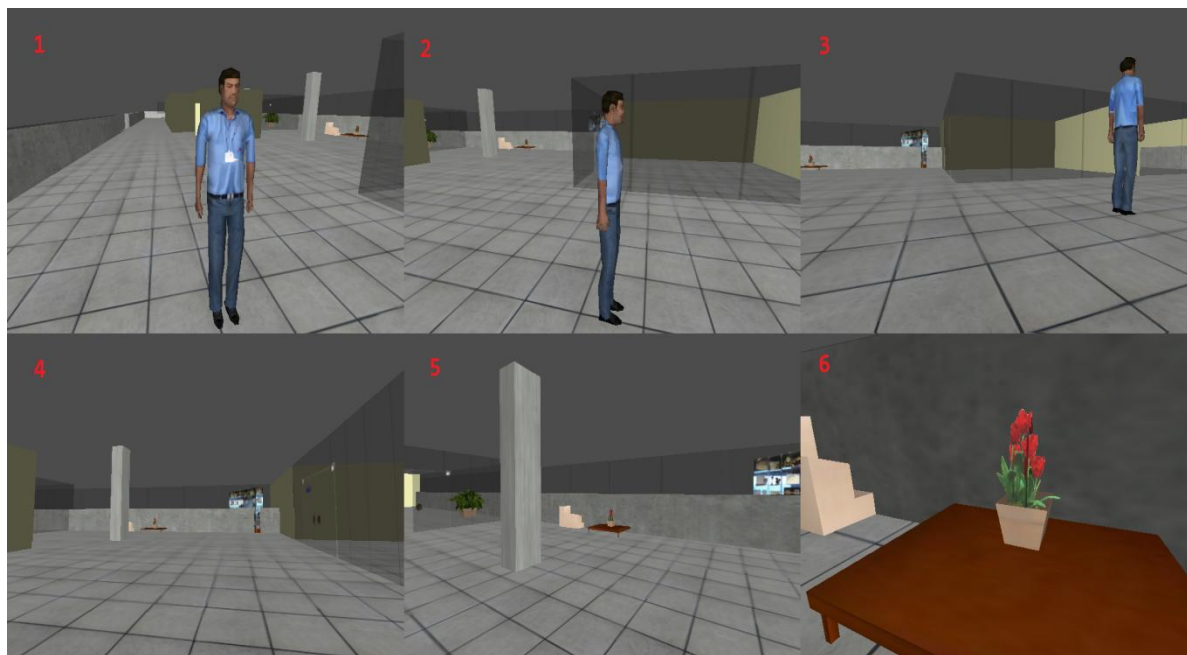
Figura 4.22 - Representação quadro-a-quadro de uma animação de câmera no Sistema de Autoria ITV

Animação de transparência

Esta animação simples, porém muito útil, torna um ou mais objetos transparentes ou opacos de maneira gradativa. Pode ser utilizado para visualizar peças internas à outras. É importante ressaltar que o objeto pode ser animado até um valor de transparência intermediário, por exemplo, 60% de transparência.

Animação embutida

O Sistema de Autoria ITV pode utilizar este tipo de animação em objetos da cena. Uma animação embutida pode ser escolhida em um mesmo objeto para ser reproduzida em um ou mais passos. O software pode ainda reproduzir a animação embutida em loop e em um intervalo de tempo específico. A figura 4.23 apresenta um exemplo de animação embutida reproduzida no sistema de autoria.

Figura 4.23 - Animação embutida sendo utilizada no Sistema de Autoria ITV

Animação de realce

Nesta animação, objetos *mesh* apresentam uma cor, escolhida pelo usuário, diferenciada e intermitente sobre a sua superfície, destacando-se dos outros objetos Este tipo de animação é muito útil em projetos de treinamento para demonstrar objetos importantes ou que fazem parte da interação do usuário. A figura 4.24 demonstra o uso deste recurso no Sistema de Autoria ITV.

Figura 4.24 - Animação de realce em objetos no Sistema de Autoria ITV

Animação de narração sonora

Esta animação nada mais é que um som reproduzido durante um passo qualquer. Foi nomeada de narração porque o som ouvido é o chamado som em duas dimensões, ou seja, permanece sem mudança no volume ou freqüência independentemente do cenário do ambiente 3D e da câmera. Devido a estas características o som é freqüentemente utilizado para narrações.

Animação de som de objeto

Esta animação representa a emissão de um som por um objeto qualquer da cena. Desta forma, o som utilizado na animação é do tipo 3D, ou seja, o usuário da ITV pode perceber a posição do objeto a partir do som, mudando-se o volume e a freqüência sonora de acordo com a visão da câmera. O Sistema de autoria ITV gerencia toda esta imersão sonora automaticamente.

4.2.6 Linha de tempo

No Sistema de Autoria ITV, cada passo possui um conjunto de linhas de tempo. As linhas de tempo desse conjunto apresentam a mesma duração de tempo entre si. Cada linha de tempo representa um objeto da cena que pode receber animações, além da linha de tempo que representa o passo em si, podendo receber animações específicas de cada tipo de objeto. A figura 4.25 mostra o conjunto de linhas de tempo de um passo qualquer de um processo ITV.
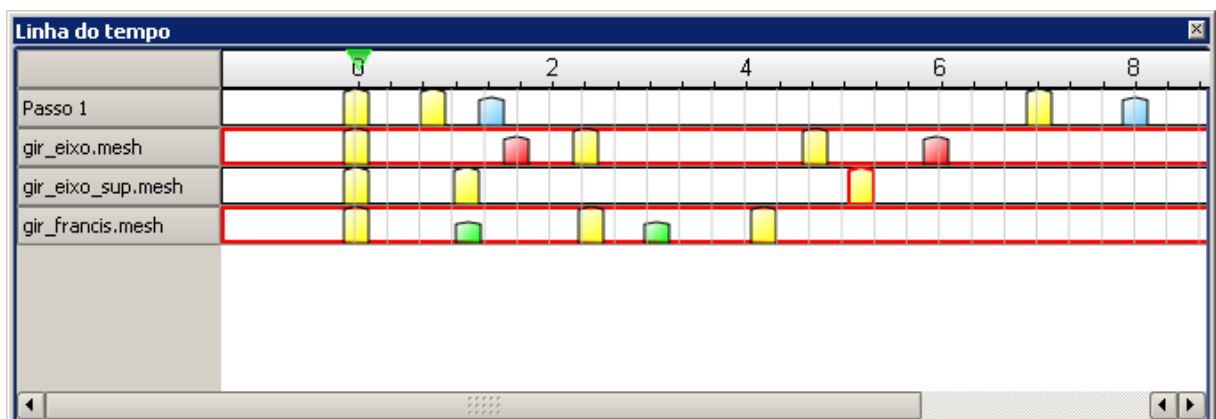
Figura 4.25 - Linhas de tempo do Sistema de Autoria ITV.

Os cursores coloridos apontando para cima no exemplo da figura 4.25 são os chamados marcadores de animação. Eles indicam a posição temporal em que algum evento da animação irá ocorrer. Por exemplo: os marcadores vermelhos na linha de tempo do objeto “gir_eixo.mesh” são marcadores de uma animação de rotação, onde o primeiro marcador indica tempo de início da rotação do objeto (1,66 segundos) e o segundo representa o tempo em que a rotação pára (7 segundos). Outro exemplo são os marcadores amarelos na linha de tempo “passo_1” indicando uma animação de câmera, onde cada marcador representa uma visão da câmera em um dado momento durante o tempo do passo.

Os marcadores podem ter sua posição alterada, através do mouse, ao longo da linha de tempo de modo que os tempos dos eventos das animações também serão alterados. Pode-se também alterar toda a animação representada pelo marcador clicando-se duas vezes nele.

Sendo assim as animações de um projeto ITV ficam totalmente representada nas linhas de tempo através dos marcadores, que podem ser distinguidos pelas cores e tamanhos específicos de cada animação que ele representa.

As linhas de tempo ainda possuem outras utilidades importantes como: o usuário pode alterar o tempo de qualquer passo da ITV, alteração que é refletida nas linhas de tempo daquele passo; Um cursor especial representa o tempo corrente da animação durante a sua reprodução; Os objetos da cena que possuem animação podem ser facilmente selecionados através das linhas de tempo que os representam; Uma descrição completa da animação pode ser visualizada pelo usuário através dos marcadores.

4.3 Outras funcionalidades

O Sistema de Autoria ITV possui ainda algumas funções que visam uma maior comodidade na criação de projetos ITV, sendo certas funções também estão presentes no modo de execução. São elas:

- Modo de visualização das animações para teste dos projetos
- Zoom rápido e automático em objetos da cena
- Os passos do projeto podem ser reaproveitados em outros momentos
- Um passo pode ter suas animações invertidas automaticamente
- Salvamento de imagens da cena 3D
- Customização da navegação pelo teclado e mouse

- Opção de utilização de um céu do tipo SkyBox em vez de fundo monocromático
- Função de desfazer e refazer ações no Sistema de Autoria ITV
- Som ambiente com customização do volume

4.4 Modos de treinamento

Durante a criação de um projeto ITV no Sistema de Autoria ITV, podem-se inserir tarefas para que o usuário da ITV final resolva-as. Por exemplo, o usuário deve acionar uma alavanca antes de abrir uma tampa. Este tipo de tarefa é de suma importância para uma forte característica do Sistema de Autoria ITV que é o treinamento.

O criador de uma ITV que contém tarefas de treinamento dispõe de funções para instruir o treinando e ao mesmo tempo avaliá-lo de forma automática durante o treinamento da ITV. Além disso, essas funções podem ser utilizadas ou não em uma mesma ITV pronta, fazendo-se uso dos modos de visualização. Para agregar uma característica de treinamento a uma função do Sistema de Autoria ITV, por exemplo, fazer com que um painel textual seja uma instrução textual de treinamento, basta marcar a caixa "Somente modos guiado e automático" na edição de uma animação, como mostra a figura 4.26.

Os modos de visualização do Sistema de Autoria ITV são baseados no estudo de (SOUSA, 2009) o qual ratifica a importância do aprendizado através de simulações em realidade virtual, além de apresentar o treinamento dividido em etapas de aprendizado crescente através dos modos de visualização.
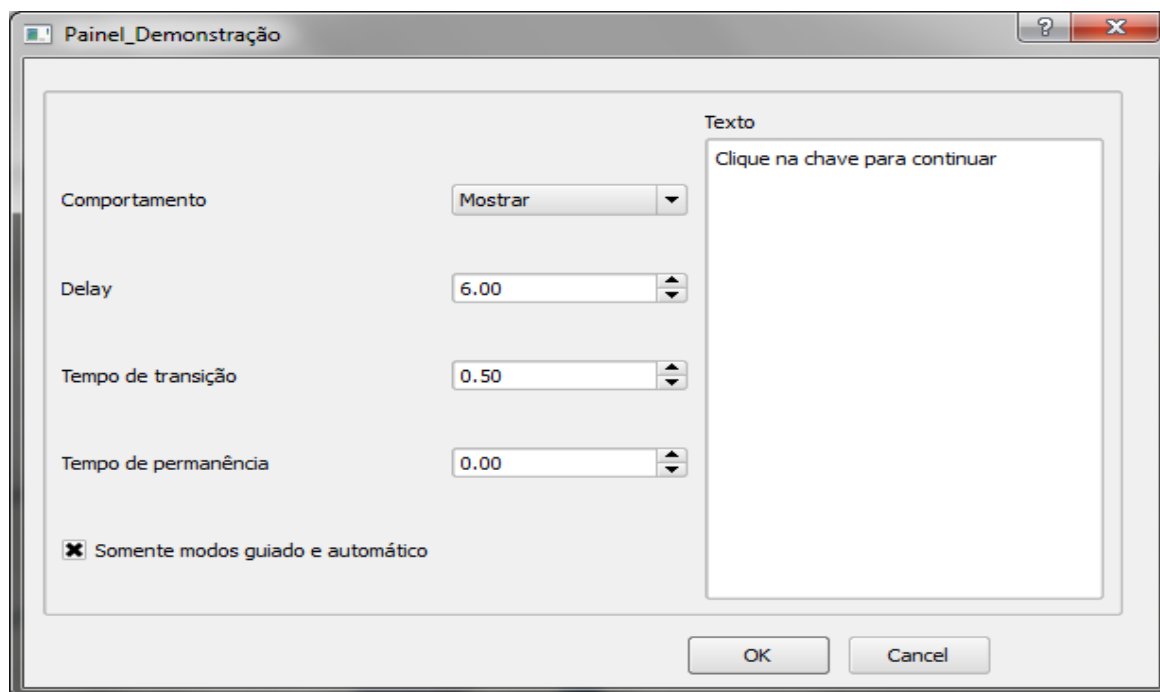
Figura 4.26 - Exemplo de função de treinamento

4.4.1 Modo Automático

Este modo apresenta o projeto ITV como uma animação de forma que o usuário da ITV se familiarize com as peças, procedimentos e instruções a respeito da instrução técnica. Como o próprio nome diz, toda a ITV será executada sem intervenção do usuário, portanto não há qualquer possibilidade de erro no treinamento por parte do usuário. Neste modo, as instruções de treinamento são apresentadas, por exemplo, o painel textual da figura 4.26 seria mostrado.

4.4.2 Modo Guiado

No modo guiado, o Sistema de Autoria ITV irá conduzir o usuário por meio de instruções textuais ou sonoras na realização dos procedimentos da ITV, sendo que a principal característica deste modo é a interação do usuário com o ambiente virtual. Desta forma, diferentemente do modo automático, o modo guiado permite que o usuário de fato realize o procedimento descrito pela ITV recebendo instruções de onde interagir com o ambiente virtual, porém assim como o modo automático, este modo também apresenta as instruções de treinamento.

A interação ocorre através do mouse, clicando-se na peça referente ao procedimento, por exemplo: para a retirada de quatro parafusos de uma tampa, deve-se utilizar a ferramenta “chave de boca” para realizar a folga deles. Então a ITV instrui o usuário a selecionar a ferramenta correta para o procedimento. O usuário então pode clicar no objeto do ambiente virtual que representa a ferramenta. O modo guiado de visualização possui também o módulo de supervisão que vai coletar todas as estatísticas de interações do usuário, a fim de avaliá-lo ao fim da execução ITV.

4.4.3 Modo Livre

O modo livre não apresenta qualquer instrução que auxilie o usuário da ITV na realização dos procedimentos, portanto, instruções como a da figura 4.26 não são mostradas neste modo, sendo assim o usuário deve estar totalmente familiarizado com a instrução técnica antes de utilizar este modo, apresentando dessa forma um maior desempenho durante a execução do treinamento da ITV.

4.4.4 Módulo de Supervisão

O módulo de supervisão está presente nos modos de visualização guiado e livre. Este módulo coleta automaticamente as interações do usuário no decorrer do treinamento da ITV, captando informações como número de tentativas e de acertos em certo procedimento da ITV.

Este módulo também gera um relatório de avaliação do usuário da ITV de treinamento. A figura 4.27 apresenta um exemplo de relatório de treinamento do Sistema de Autoria ITV.

Relatório

Usuário Nome
Wed Jul 28 2010

Taxa de acerto: 33.3333%
Tentativas: 6

24- Acionamento da chave para controle remoto --> 25-Acionamento da chave para controle remoto - Animação
Tentativas: 3
Acertos: 1
Erros: 2
36- Validação da abertura total da comporta --> 37-Validação da abertura total da comporta - Fade
Tentativas: 3
Acertos: 1

Sumário | Mostrar número de tentativas | Mostrar somente os passos com tentativas

Salvar | Fechar

Figura 4.27 - Relatório gerado pelo módulo de supervisão da ITV

5 ESTUDO DE CASO: ITV DE PARTIDA DE UMA UNIDADE HIDROGERADORA

Uma unidade hidrogeradora ou UHG é composta por diversos equipamentos, os quais se completam para cumprir a tarefa de gerar energia elétrica a partir da energia cinética da água. Entre esses equipamentos estão: o gerador, mancais, turbina, distribuidor, sistema de excitação, regulador de velocidade, comporta de tomada d'água, transformadores, regulador de tensão, dentre outros. Estes equipamentos têm uma relação direta com a partida da UHG. Na Usina Hidroelétrica de Tucuruí existe um modo de partida da UHG chamado de partida automática, o qual permite que um operador monitore todo o processo principal através de um computador, coletando e enviando comandos para a UHG através de sensores e relés instalados por todos os equipamentos da unidade.

A seqüência de partida de uma UHG é um processo bem definido e repetido com certa freqüência na Usina de Tucuruí., porém não é facilmente observado fora dos monitores de computador. A intenção da criação de uma ITV que descreva este processo é apresentar todas as alterações ocorridas nos equipamentos da UHG mostrando os próprios equipamentos, o que seria impossível de ser feito na UHG real, por exemplo, visualizar o sistema distribuidor da máquina em funcionamento seria impossível sem o uso dos recursos de realidade virtual.

Contudo, ao mesmo tempo em que o processo é visto de uma perspectiva única através da simulação 3D, esta ITV não deixa de apresentar o processo pela visão do operador real da máquina, o qual controla o processo de partida através de computadores. Esta ITV também conta com os modos de visualização para treinamento, onde o usuário deverá tomar o papel dos diferentes operadores que participam do processo de partida da UHG.

5.1 Roteiro resumido da ITV

A ITV descrita resumidamente nesta seção é composta de oitenta e dois passos totalizando uma duração de tempo de dezessete minutos se for visualizada no modo automático de treinamento.

O primeiro passo apresenta o título da ITV e seu propósito, mostrando também uma panorâmica de toda a Usina Hidrelétrica de Tucuruí através de uma animação de câmera do Sistema de Autoria ITV. Neste passo utilizou-se também uma animação de painel textual na

cena para apresentar o título da ITV. A figura 5.1 mostra uma imagem capturada durante a animação deste passo.
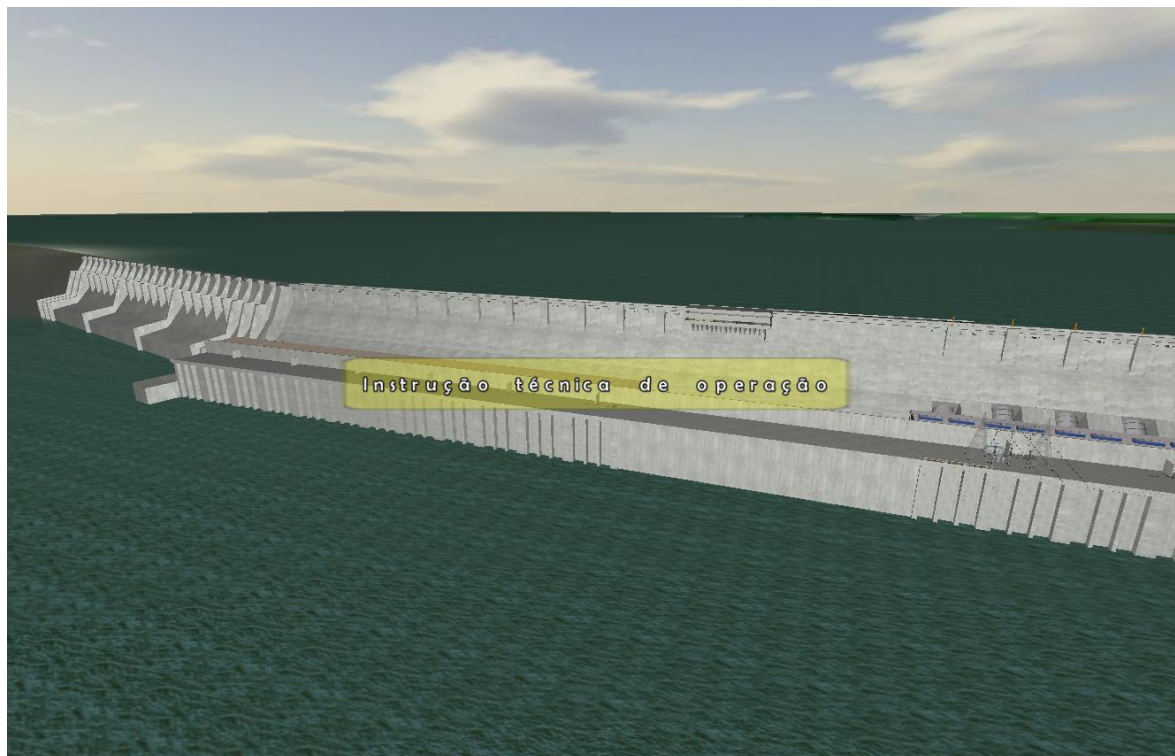
Figura 5.1 - Primeiro passo da ITV de partida

O passo 14 apresenta uma das pré-condições de partida sendo atendidas, neste caso ocorre a abertura da válvula principal do sistema de regulação para normalização desse sistema. A abertura é feita manualmente por um operador funcionário da usina, o qual utiliza uma ferramenta para facilitar o processo. Muitas animações foram utilizadas neste passo, sendo uma animação embutida para animação do esqueleto do avatar que representa o operador, animações de translação para representar a locomoção do avatar, animação de painel textual na cena e animação de rotação para simulação do uso da ferramenta, além é claro de animações de câmera para a correta visualização do processo. A figura 5.2 apresenta este processo na ITV.

Figura 5.2 - Passo que mostra a realização de tarefas de pré-condição de partida.

Durante o passo 19, o COU, Centro de Operações da Usina, é apresentado. Neste local um operador irá controlar e monitorar acontecimentos do processo de partida da UHG. A figura 5.3 mostra o COU e seus equipamentos de controle e monitoração que serão utilizados pelo funcionário e pelo usuário da ITV durante a simulação da partida.

Figura 5.3 - Imagem do passo 19 mostrando o COU

O acionamento do equipamento chamado “centralina”, que controla a abertura e fechamento da comporta da tomada d’água, é representado na ITV com animações de transparência, animações de realce de objetos e também com a interação do usuário representando a ativação do controle automático da centralina pelo operador. A figura 5.4 representa estes acontecimentos durante o passo 24.

Figura 5.4 - Acionamento da centralina representada no Sistema de Autoria ITV

No passo 48, após ter verificado e alcançado todas as pré-condições para o início do processo de partida, tem-se outra interação do usuário com ambiente virtual, desta vez para o comando de partida da máquina propriamente dita. O comando é feito através da interface de monitoração das Unidades Hidrogeradoras no COU e é representado na ITV da forma mais real possível, inclusive com o usuário clicando na tela para o acionamento do comando, como mostra a figura 5.5.
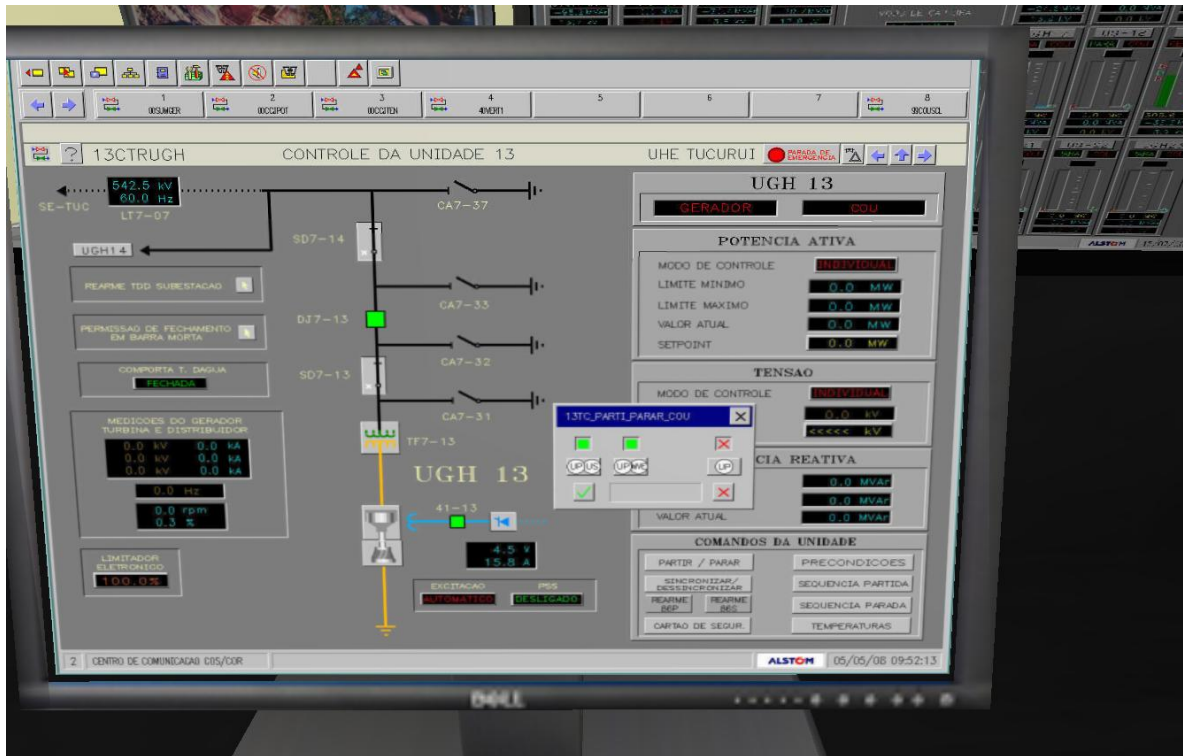
Figura 5.5 - Interação do usuário com a interface virtual do COU

O comando de partida desencadeia uma série de atividades essenciais na UHG. Dentre algumas destas atividades estão os acionamentos das bombas responsáveis pela circulação de óleo nos mancais e no regulador de velocidade da turbina. A circulação do óleo é representada por setas animadas percorrendo todo o caminho do óleo através da UHG. O Sistema de Autoria pode facilmente utilizar este tipo de animação da textura do objeto. A figura 5.6 mostra este tipo de representação na ITV durante o passo 53.
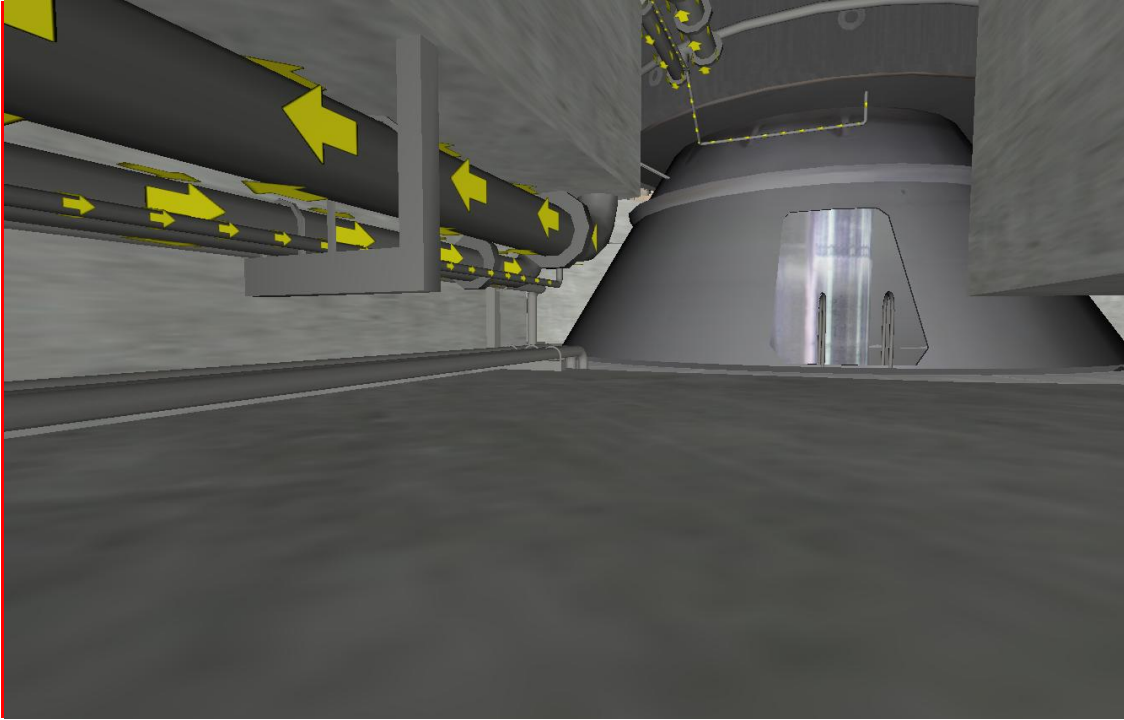
Figura 5.6 - Animação de textura sendo utilizada para representar o fluxo de óleo

No passo 64 ocorre a primeira abertura das palhetas diretrizes da turbina, esta abertura pode ser observada de maneira única na ITV, já que de outro modo seria impossível acessar aquele ponto, que está totalmente inundado de água. A ITV utiliza animações de rotação para controlar a abertura das 24 palhetas desta UHG, além disso, utiliza também animação de transparência para abstrair outras peças não importantes para a visualização do processo. A figura 5.7 mostra a região da UHG conhecida como caixa espiral, já com as devidas transparências aplicadas; é nesta região que ocorre a abertura das palhetas diretrizes.
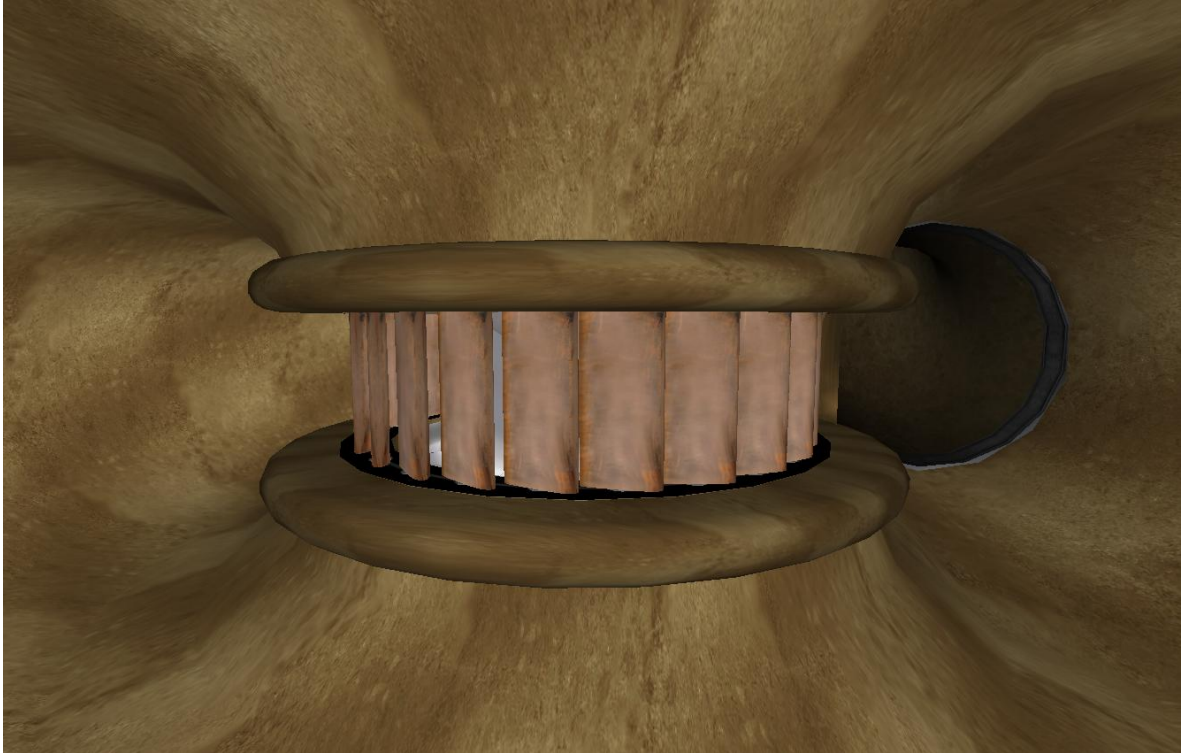
Figura 5.7 - Abertura das palhetas diretrizes

No passo 77 ocorre a geração de energia propriamente dita. O rotor da UHG começa a girar em uma velocidade perfeitamente controlada depois de algumas intervenções do operador da usina, intervenções que também são simuladas na ITV. Esta ITV de operação mostra o giro do rotor através de animações de rotação, representa a geração de energia através de texturas animadas e abstrai algumas peças para melhor apreciação do processo através de animações de transparência. A figura 5.8 mostra uma imagem que representa este passo.
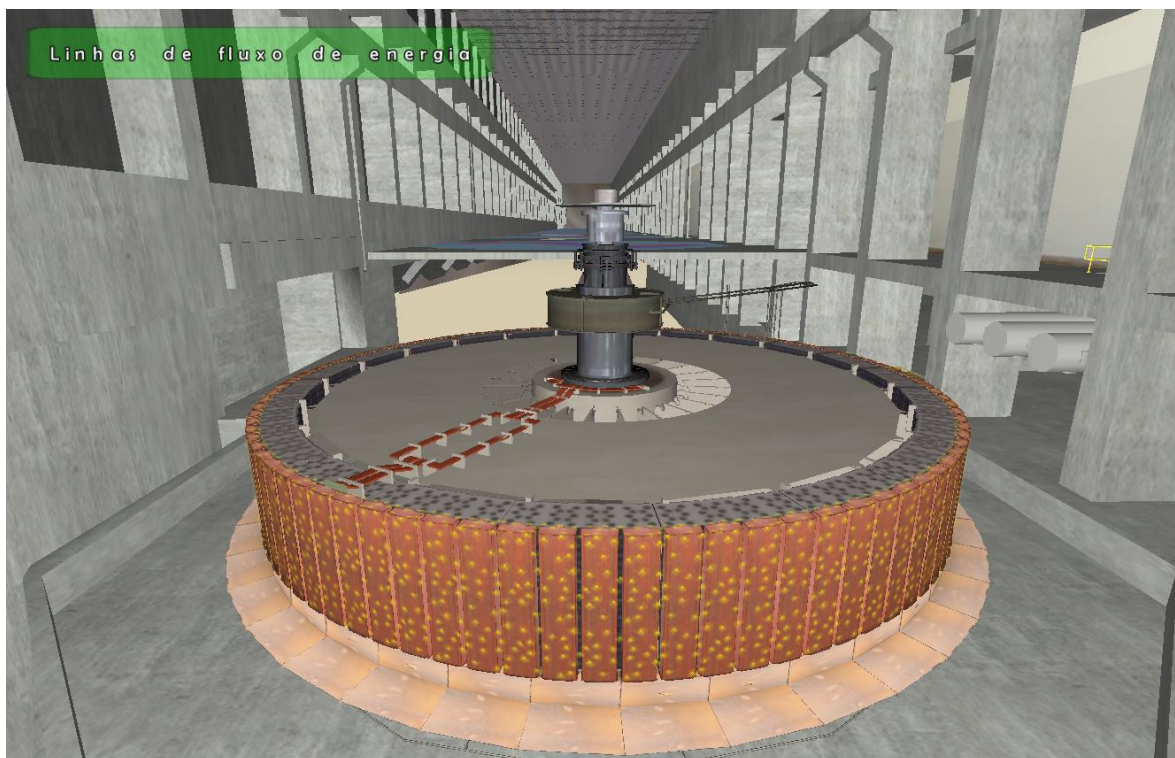
Figura 5.8 - Imagem da ITV de partida representando o início da geração de energia.

Os passos finais da ITV de partida da UHG mostram o caminho do fluxo de energia desde o rotor do gerador até o transformador e daí para fora da usina. Animações de câmera são utilizadas para navegar pelo fluxo e mostrar boa parte da usina. No fim da ITV, caso o usuário tenha visualizado-a em um dos modos de treinamento (guiado ou livre), é apresentado um relatório de avaliação gerado pelo Sistema de Autoria ITV, o qual contém essencialmente os erros e acertos do usuário durante a simulação da ITV de partida.

5.2 Avaliação da ITV

O Sistema de Autoria ITV pôde ser avaliado em 2010, antes da conclusão do projeto. Esta avaliação foi realizada pelos próprios usuários finais do software. Neste caso, quarenta funcionários da Eletronorte envolvidos com a Usina Hidoelétrica de Tucuruí utilizaram ITV's totalmente criadas no Sistema de Autoria ITV e puderam expor suas opiniões sobre a relevância do projeto para o uso na rotina de trabalho. Dentre os funcionários avaliadores estavam técnicos de manutenção elétrica, técnicos de manutenção eletrônica, operadores da

usina, engenheiros de automação e gerentes. O estudo completo desta avaliação pode ser visto em (JUNIOR e FILHO, 2010)

O primeiro gráfico mostrado na figura 5.9 representa a avaliação da relevância do treinamento virtual no Sistema de Autoria ITV para o treinamento real na usina. Pode-se perceber que a grande maioria dos avaliadores entendeu que o Sistema de Autoria ITV é bastante adequado neste aspecto de treinamento.
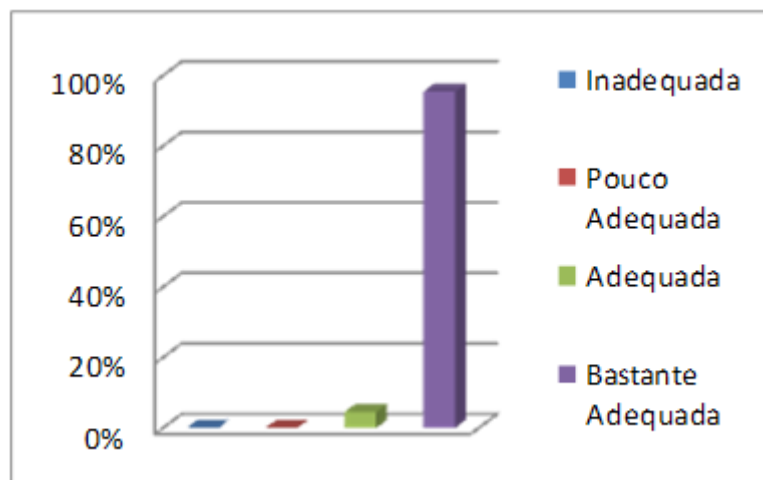
Figura 5.9 - Avaliação da adequação do treinamento em RV para o treinamento real

Avaliou-se também a usabilidade do sistema de modo a demonstrar a facilidade no uso do programa e sua interface. A figura 5.10 mostra que a maioria dos avaliadores julgou a interface como fácil de usar. Percebe-se que uma parte dos avaliadores julgou como difícil o uso do programa, contudo, após uma entrevista com este grupo de avaliadores, constatou-se que eles não possuíam contato com sistemas computacionais na rotina de trabalho. A boa recepção da usabilidade do programa é importante em um sistema de criação que vai ser utilizado por usuários não familiarizados com a área de computação gráfica. Vale lembrar que a usabilidade avaliada aqui também engloba a interação do usuário com o ambiente virtual do Sistema de Autoria ITV, seja por meio do mouse ou do teclado em ações de manipulação dos objetos da cena.
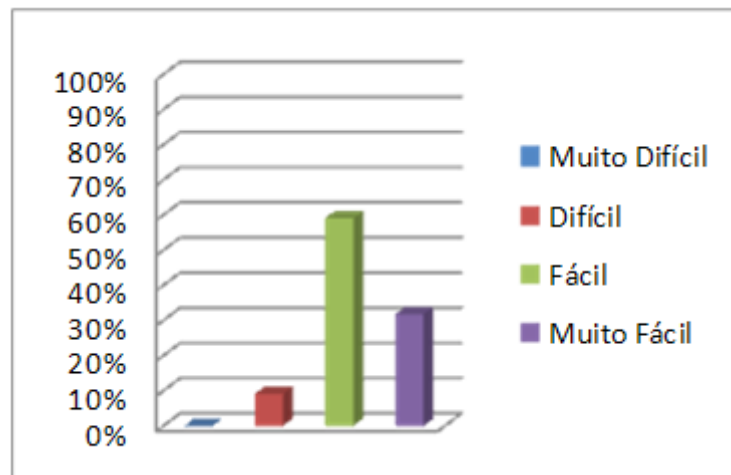
Figura 5.10 - Nível de usabilidade do Sistema de Autoria ITV

Outra questão avaliada foi a quantidade e qualidade do conhecimento proporcionado com o treinamento pelo Sistema de Autoria ITV em relação ao treinamento comum. A figura 5.11 mostra que a grande maioria julgou que o treinamento em RV proporcionado pela ITV apresenta mais informações que o treinamento tradicional. Este resultado foi previsível visto que a liberdade de visualização das peças envolvidas no processo é muitas vezes maior que a visualização do processo real.
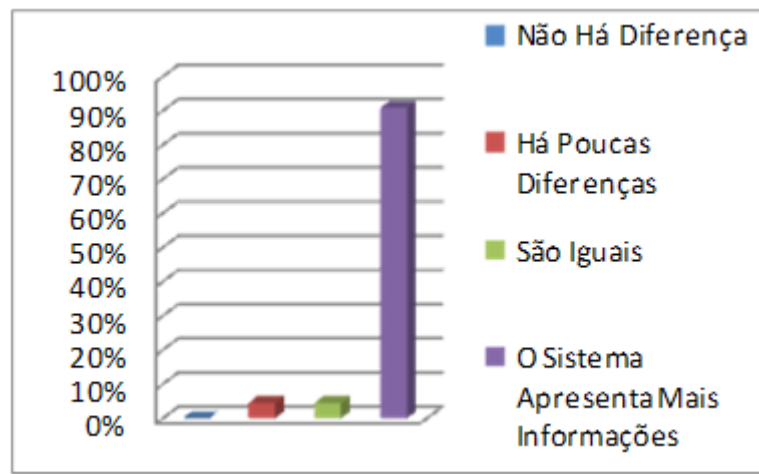
Figura 5.11 - Comparação dos conhecimentos adquiridos entre os modos de treinamento

Outro quesito avaliado pelos funcionários da Eletronorte foi o grau de facilidade com que as informações são assimiladas em um processo visto em RV em comparação ao processo visto na maneira tradicional. A figura 5.12 apresenta resultado desta avaliação a qual todos os

avaliadores julgaram que o grau de retenção da informação é superior no treinamento pelo Sistema de Autoria ITV.
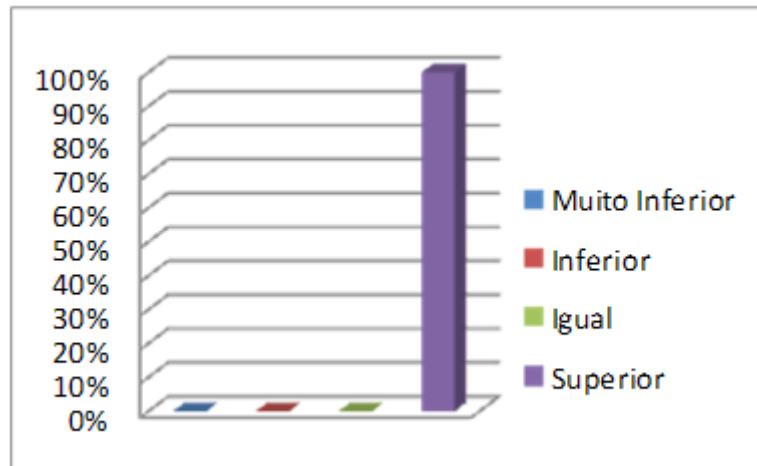
Figura 5.12 - Comparação da retenção de informações através da ITV

6 CONCLUSÃO E TRABALHOS FUTUROS

Esta dissertação apresentou um sistema de software completo para a criação de animações, simulações e treinamentos em realidade virtual. O sistema apresenta uma interface consistente que permite que um usuário sem conhecimentos de programação de computadores possa criar procedimentos técnicos denominados de instruções técnicas virtuais. O sistema também garante uma metodologia de treinamento baseado em estudos anteriores de treinamento em RV.

Os primeiros capítulos da dissertação apresentaram tecnologias semelhantes ao trabalho exposto explicando a importância dessas tecnologias para a construção das aplicações em realidade virtual. Os capítulos seguintes explanaram com detalhes a arquitetura envolvida no desenvolvimento do Sistema de Autoria ITV e as bibliotecas de software utilizadas no desenvolvimento. Os capítulos finais apresentam o software pronto e suas funcionalidades, demonstrando o seu uso através de um estudo de caso, além de apresentar uma pesquisa que avaliou o Sistema de Autoria ITV e sua importância.

A interface simples e baseada em outros sistemas de software populares facilita o aprendizado e o uso do Sistema de Autoria ITV. Além disso, o uso de do mouse e teclado no ambiente virtual para a navegação e construção de animações possibilita um processo de criação intuitivo e rápido. O uso das linhas de tempo também agiliza a edição das animações e manipulação dos objetos envolvidos na cena.

O conceito de passos utilizado no Sistema de Autoria ITV provê ao usuário uma construção de simulações de forma estruturada, exatamente como ocorre no processo real, onde cada passo representa uma tarefa simples a ser feita para a conclusão do processo como um todo. Esta característica representa um grande avanço no uso de simulações e treinamentos em RV em relação ao treinamento convencional discutido também nesta dissertação.

Observou-se a grande evolução do sistema de software exposto quando comparado ao projeto anterior UGV, buscando como principal característica a liberdade para construir qualquer ITV com facilidade e rapidez, equiparando-se a sistemas de autoria utilizados em diferentes empresas e setores da indústria pelo mundo, contando ainda com características diferenciadas como o processo de construção livre de codificação e *scripts* e o módulo

autônomo de supervisão de treinamento, possibilitando um treinamento mais confiável e consistente.

O Sistema de Autoria ITV encontra-se em fase de distribuição onde está sendo implantado na Usina Hidroelétrica de Tucuruí para utilização dos funcionários da Eletronorte. A implantação já conta com uma base de recursos de mais de mil e quinhentos modelos 3D representando peças da Usina e mais dez ITV's prontas representando processos utilizados com freqüência na Usina.

Dentre algumas melhorias e novas funcionalidades que são interessantes ao Sistema de Autoria ITV pode-se destacar a inclusão de informações textuais das peças envolvidas na simulação. Estes textos poderiam ser informações com peso das peças, ano de fabricação, última manutenção ou detalhamento dos materiais utilizados na fabricação da peça que seriam úteis para melhorar a qualidade da representação do mundo real, já que nem todas as informações podem ser expostas somente no ambiente virtual 3D.

Outra melhoria de grande utilidade seria a possibilidade de gravação de vídeos das simulações construídas no Sistema de Autoria ITV, de modo que se pudesse apresentar a animação em diferentes dispositivos como celulares ou aparelhos de DVD. Esta gravação deveria estar embutida no Sistema de Autoria ITV e permitiria uma distribuição tão simples que o vídeo gerado poderia ser facilmente distribuído em sites na Internet já que não necessitaria do Sistema de Autoria ITV para a visualização da animação. É claro que uma característica importante seria perdida: a interação do usuário com o ambiente virtual nos modos de treinamento.

Pode-se também pensar em uma distribuição multiplataforma, visto que o Sistema de Autoria ITV pode ser executado apenas em plataforma Windows. Esse ambiente multiplataforma poderia incluir até mesmo dispositivos portáteis ou execução embutida em tecnologia Flash para distribuição das simulações na web.

Uma grande evolução do Sistema de Autoria ITV seria a criação de simulações não determinísticas, ou seja, a simulação começa de uma forma porém pode seguir por caminhos diferentes e finalizar de formas diferentes. Este tipo de simulação é interessante para a simulação de erros em processos. Por exemplo, no processo de partida de uma UHG, descrito nesta dissertação, pode haver erros que impossibilitem continuar o processo. O Sistema de Autoria simularia esses erros e permitiria que o usuário interagisse com o ambiente virtual a fim de contornar o problema e retomar o processo normalmente.

7 REFERENCIAS BIBLIOGRÁFICAS

ANARK. www.anark.com, 2010. Disponivel em: <http://www.anark.com/media/>.

ANDERSON, E. F. et al. The Case for Research in Game Engine Architecture. **Conference on Future Play**, 2008.

AUTODESK. Autodesk, 2010. Disponivel em: <http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13567410>.

BARATA, P. N. Virtual Technical Instructions of Maintenance and Operation of a HydraulicGenerating Unit. **Symposium on Virtual and Augmented Reality**, Porto Alegre, 2009.

BERGEN, G. **Collision Detection in Interactive 3D Environments**. Chicago: Morgan Kaufmann, 2003.

BLANCHETTE, J.; SUMMERFIELD, M. **C++ GUI Programming with Qt 4**. 2ª edição. ed. [S.l.]: Prentice Hall, 2008.

BORLAND. Borland, 2010. Disponivel em: <http://www.borland.com/>.

DAMASCENO, R. R. **Concepção do jogo educativo "A Revolta da Cabanagem": enredo, cenário, interfaces, jogabilidade e áudio.** UFPA. Belém. 2009.

DEGIOVANI, M. **GLScene Passo-a-Passo**. [S.l.]: [s.n.], 2003.

DEGOES, J. **3D Programming with C++**. [S.l.]: Paraglyph Press, 1999.

DEITEL, H. M. **C ++ Como Programar**. 6ª edição. ed. [S.l.]: Prentice Hall, 2006.

DYKES, L.; TITTEL, E. **XML For Dummies**. 4ª edição. ed. [S.l.]: For Dummies, 2005.

EBERLY, D. **3D Game Engine Design:** A Practical Approach to Real-time Computer Graphics. [S.l.]: Morgan Kaufmann, 2000.

ELETRONORTE. Eletronorte, 2010. Disponivel em: <www.eln.gov.br>.

EVENS, M.; MICHAEL, J. **One-on-One Tutoring by Humans and Computers**. [S.l.]: Lawrence Erlbaum Associates, 2005.

FMOD. FMOD, 2010. Disponivel em: <www.fmod.org>.

GREGORY, J. **Game Engine Architecture**. [S.l.]: A K Peters, 2009.

HESS, R. **The Essential Blender:** Guide to 3D Creation with the Open Source Suite Blender. [S.l.]: No Starch Press, 2007.

HOULMANN, F.; METZ, S. **High Dynamic Range Rendering in OpenGL**. [S.l.]. 2009.

JUNIOR, A. P.; FILHO, M. R. Instrução Técnica Virtual - Válvula Detectora de Pressão. **Simpósio Brasileiro de Sistemas Elétricos**, 2010.

JUNKER, G. **Pro OGRE 3D Programming**. [S.l.]: Apress, 2006.

KIAMEH, P. **Power generation handbook**. [S.l.]: McGraw Hill Professional, 2002. Disponivel em: <http://pt.wikipedia.org/wiki/Turbina_hidr%C3%A1ulica>.

KODU. Kodu, 2009. Disponivel em: <http://fuse.microsoft.com/kodu/>.

LARV. LaRV, 2010. Disponivel em: <www.larv.ufpa.br>.

LAW, A.; KELTON, W. D. **Simulation Modeling and Analysis**. 3ª edição. ed. [S.l.]: McGraw-Hill Science/Engineering/Math, 1999.

MICROSOFT. Power Point, 2010. Disponivel em: <http://office.microsoft.com/en-us/powerpoint/>.

NETTO, A. A. D. O.; TAVARES, W. R. **Introdução à Engenharia de Produção**. [S.l.]: Visual Books , 2006.

NGRAIN. NGRAIN, 2010. Disponivel em: <www.ngrain.com/products/producer.html>.

NITSCHKE, B. **Professional XNA Game Programming:** For Xbox 360 and Windows. [S.l.]: Wrox, 2007.

OGRE. Ogre, 2010. Disponivel em: <www.ogre3d.org>.

ONG, J.; RAMACHANDRAN, S. Intelligent Tutoring Systems: The What and the How. **ASTD: The American Society for Training & Development**, 2000.

OPCODE. www.codercorner.com/Opcode.htm, 2008. Disponivel em: <www.codercorner.com/Opcode.htm>.

PAMPLONA, A. Um sistema de Realidade Virtual para Treinamento de Manutenção e Estudo de uma Unidade Hidrelétrica de Energia. **Congresso Argentino de Ciências da Computação**, 2006.

RAGE. RAGE, 2010. Disponivel em: <www.mobygames.com/game-group/3d-engine-rage>.

RANDIMA, F. **GPU Gems:** Programming Techniques, Tips and Tricks for Real-Time Graphics. 1ª edição. ed. [S.l.]: Addison-Wesley Professional, 2004.

ROGER, P. **Engenharia de Software**. 6ª edição. ed. [S.l.]: McGraw-Hill, 2006.

SERVICES, F. I. **Shareable Content Object Reference Model**. [S.l.]: Faulkner Information Services, 2001.

SOUSA, M. A. **Uma nova abordagem de treinamento de manutenção em uma unidade hidrogeradora elétrica, utilizando tecnologias de realidade virtual desktop**. UFPA. Belém. 2009.

SPOLSKY, J. **User Interface Design for Programmers**. [S.l.]: Apress, 2001.

TATARINOV, A. **Instanced Tesselation in DirectX10**. Game Developers Conference. [S.l.]: [s.n.]. 2008.

TEAM, A. C. **Adobe Flash CS3 Professional Classroom in a Book**. [S.l.]: Adobe Press, 2007.

TORQUE. Torque, 2009. Disponivel em: <http://www.torquepowered.com/products/torque-3d>.

UNITY. Unity, 2010. Disponivel em: <http://unity3d.com/unity/features/>.

UNREAL. Unreal, 2010. Disponivel em: <www.unrealtechnology.com>.

VILHENA, P. R.; FILHO, M. R. Utilizando técnicas de realidade virtual para o desenvolvimento de programa computacional que objetiva a representação da partida, parada e funcionamento de uma unidade hidrogeradora integrada em um sistema elétrico. **Seminário Nacional de de Produção e Transmissão de Energia Elétrica**, 2007.

WEISSTEIN, E. W. Quaternion, 2010. Disponivel em: <http://mathworld.wolfram.com/Quaternion.html>.

WORLDS, T. Thinking Worlds, 2009. Disponivel em: <www.thinkingworlds.com>.